Front. Eng. Manag. 2018, 5(4): 515–523
https://doi.org/10.15302/J-FEM-2018044

RESEARCH ARTICLE

Panos M. PARDALOS, Mahdi FATHI

A discussion of objective function representation methods in global optimization

Abstract Non-convex optimization can be found in several smart manufacturing systems. This paper presents a short review on global optimization (GO) methods. We examine decomposition techniques and classify GO problems on the basis of objective function representation and decomposition techniques. We then explain Kolmogorov's superposition and its application in GO. Finally, we conclude the paper by exploring the importance of objective function representation in integrated artificial intelligence, optimization, and decision support systems in smart manufacturing and Industry 4.0.

Keywords global optimization, decomposition techniques, multi-objective, DC programming, Kolmogorov's superposition, space-filling curve, smart manufacturing and Industry 4.0

1 Global optimization (GO) methods

Global non-convex programs can be solved using several approaches according to recent advances in GO literature (Pardalos and Rosen, 1986; Pardalos, 1991; Bomze et al., 1997; Pardalos and Wolkowicz, 1998; Horst et al., 2000; Nowak, 2005; Floudas and Pardalos, 2013; Horst and Pardalos, 2013; Floudas and Pardalos, 2014). These approaches can be divided into exact methods that can find and verify global solutions and heuristic methods, which only seek global solutions without checking optimality. Heuristics achieve a critical function in the optimization of large-scale non-convex problems and can be applied to provide upper bounds for global optimum, generate cuts and relaxations, and partition feasible sets. Approximation algorithms are kinds of heuristics, wherein performance guarantee is considered estimated error (Fisher, 1980; Hochbaum et al., 1999; Ausiello et al., 2012; Vazirani, 2013). MIP approximation techniques work by approximating univariate functions to piecewise linear function with a performance guarantee for MINLP method. Goemans and Williamson (1995) solved a quadratic binary program using the MaxCut heuristic as first approximation algorithm.

In GO, an algorithm is called finite if it obtains and verifies a global solution in a finite number of step. The exact methods are finite in finding and verifying solution. Moreover, simplex, active set, and enumeration methods are finite for solving LPs, convex QPs, and bounded integer or concave problems. However, interior point and solution methods for SQP as a nonlinear convex program are not finite.

All GO methods create a rough model of the program for finding global solutions. A GO method is called a *sampling heuristic* if the method uses a crude model based on a finite set of points. The considered regions of interest in *sampling heuristic* methods are bounded set. The distribution of points in this region is usually denser and should consider random behavior to obtain all possible solutions. In the continuous feasible region, the possible random sample is infinite, and a GO solution is not guaranteed. Moreover, the sample can prove that the method converges with probability that is arbitrarily close to 1. A GO method is called a *relaxation-based method* if the method uses relaxation as a crude model, such as a mathematical model, which is easier to solve than the original problem. The crude model influences the problem description. Modeling the problem in an aggregated form is efficient for sampling heuristics with few variables and a simple, feasible set in a disaggregated form for relaxation-based method with objective functions and constraints that can be relaxed.

Relaxation-based heuristics are classified into three relaxation-based methods classes, which include *branch-and-bound methods.* This method divides the GO problem

Received May 2, 2018; accepted August 2, 2018

Panos M. PARDALOS (✉), Mahdi FATHI
Department of Industrial Engineering and Systems Engineering, University of Florida, Gainesville, FL 116595, USA
E-mail: pardalos@ise.ufl.edu

into subproblems based on partitioning of the feasible set. S*uccessive relaxation methods* successively improve an initial relaxation without dividing it into subproblems. *Heuristics* retrieve potential solutions from a given relaxation without modifying the relaxation.

The MINLP solver technology should be further developed, and additional details on GO (Pardalos and Rosen, 1987; Pintér, 1996; Horst et al., 2000; Neumaier, 2004; Schichl, 2010; Horst and Pardalos, 2013; Horst and Tuy, 2013;), MINLP methods (Floudas et al., 1989; Grossmann and Kravanja, 1997; Grossmann, 2002; Tawarmalani and Sahinidis, 2002; Floudas, 2013), and sampling heuristics (Torn and Zilinskas, 1989; Boender and Romeijn, 1995; Strongin and Sergeyev, 2000) should be identified. In summary, GO methods can be classified as follows:

- **Sampling heuristics**: 1) Ultistar (Strongin and Sergeyev, 2000), 2) Clustering method (Becker and Lago, 1970; Dixon and Szegö, 1974; Torn and Zilinskas, 1989), 3) Evolutionary algorithm (Forrest, 1993), 4) Simulated annealing (Metropolis et al., 1953; Kirkpatrick et al., 1983; Locatelli M, 2002), 5) Tabu search (Glover and Laguna, 1997; Mart et al., 2018), 6) Statistical GO (Mockus J, 2012), 7) Greedy randomized adaptive search procedure (Resende and Ribeiro, 2003; Hirsch et al., 2007)
- **Branch-and-bound methods**: 1) Branch-and-bound (Smith and Pantelides, 1996; Vaidyanathan and El-Halwagi, 1996; Smith and Pantelides, 1999; Horst and Tuy, 2013), 2) Branch-and-cut (Padberg and Rinaldi, 1991), 3) Branch-and-reduce (Sahinidis, 1996), 4) Branch-and-price, 5) Branch-cut-and-price, 6) Branch-and-infer (Van Hentenryck et al., 1997; Bliek, 1998; Boddy and Johnson, 2002; Sellmann and Fahle, 2003; Hooker, 2011).
- **Successive approximation method**: 1) Extended cutting-plan method (Westerlund and Pettersson, 1994, 1995; Westerlund et al., 2001), 2) Generalized bender decomposition (Geoffrion, 1972; Floudas et al., 1989; Paules and Floudas, 1989), 3) Outer approximation (Duran and Grossmann, 1986; Kocis and Grossmann, 1987; Viswanathan and Grossmann, 1990; Fletcher and Leyffer, 1994; Zamora and Grossmann, 1998a, 1998b; Grossmann, 2002; Kesavan et al., 2004), 4) Logic-based approach (Türkay and Grossmann, 1996; Vecchietti and Grossmann, 1999), 5) Generalized cross decomposition (Holmberg, 1990), 6) Successive semidefinite relaxation (Lasserre, 2001; Henrion and Lasserre, 2002; Kojima et al., 2003), 7) Lagrangian and domain cut method (Li et al., 2009).
- **Relaxation-based heuristics**: 1) Rounding heuristics (Mawengkang and Murtagh, 1986; Goemans and Williamson, 1995; Burkard et al., 1997; Zwick, 1999), 2) Lagrangian heuristics (Holmberg and Ling, 1997; Nowak and Römisch, 2000), 3) Deformation heuristics (Moré and Wu, 1997; Schelstraete et al., 1999; Alperin and Nowak, 2005), 4) MIP approximation (Neumaier, 2004), 5) Successive linear programming (Palacios-Gomez et al., 1982).

2 Decomposition theory

Large-scale problems can be solved by splitting them into subproblems, which are coupled by a master problem either in parallel or in sequence. The Dantzig–Wolfe decomposition employs separability to decompose a GO problem to subproblems; this method is one of the first decomposition approaches for linear programming that could be optimized in parallel (Dantzig and Wolfe, 1960). This method considers dual problem as a master problem, which coordinates the solutions and iterative modifications of the subproblems. The extension of Dantzig–Wolfe decomposition was applied to the nonlinear convex problem, and the Lagrangian dual is solved by using the cutting plane method. Details regarding decomposition methods in convex and non-convex GO problems are found in (Kelly et al., 1998; Bertsekas, 1999; Horst et al., 2000; Babayev and Bell, 2001; Svanberg, 2002; Palomar and Chiang, 2006; Zhang and Wang, 2006; Boyd et al., 2007; Chiang et al., 2007; Zheng et al., 2013; Rockafellar, 2016; Rahmaniani et al., 2017; Nowak et al., 2018). In general, decomposition techniques can be classified into dual and primal decomposition methods.

2.1 Primal decomposition

The following program with objective function is considered:

$$\left\{ max_{y,x_i} \sum_i f_i(x_i);\ \text{subject to}: x_i \in X_i \right\}, \qquad (1)$$

where $\forall i\ A_i x_i \leqslant y$, and $y \in Y$. Primal decomposition can be applied wherever a coupling variable is set to a fixed value. Thereafter, the GO problem is decoupled into several subproblems for each i as:

$$\{ max_{x_i}\ f_i(x_i);\ \text{subject to}: x_i \in X_i,\ A_i x_i \leqslant y \}. \qquad (2)$$

The master problem updates the coupling variable by solving:

$$\left\{ max_y \sum_i f_i^*\ (y);\ \text{subject to}\ :\ y \in Y \right\}, \qquad (3)$$

where $\lambda_i f_i^*(y)$ is the optimal objective value in (2). Therefore, Problems (2) and (3) are convex optimization problems if Problem (1) is convex. The gradient method solves Problem (3). Therefore, the optimal Lagrange multiplier, $\lambda_i^*(y)$ in (2), the subgradient for each $f_i^*(y)$ obtained by $s_i(y) = \lambda_i^*(y)$), and Problem (2) can be solved by $\boldsymbol{y}$, where $s(y) = \sum_i s_i(y) = \sum_i \lambda_i^*(y)$ is the global subgradient.

2.2 Dual decomposition

Dual decomposition is suitable when a coupling constraint and its relaxation exist. The GO problem is divided into several subproblems.

$$\left\{max_{\boldsymbol{x}_i}\sum_i f_i(\boldsymbol{x}_i);\ \text{subject to}: \boldsymbol{x}_i \in \boldsymbol{X}_i,\ \forall i\sum_i \boldsymbol{h}_i(\boldsymbol{x}_i)\leqslant \boldsymbol{c}\right\}. \tag{4}$$

The following equation is obtained by applying Lagrangian relaxation to the coupling constraint in Problem (4):

$$\left\{max_{x_i}\sum_i \mathrm{f}_i(x_i)\ -\lambda^{\mathrm{T}}\left(\sum_i h_i(x_i)\text{-}c\right);\right.$$

$$\left.\text{subject to}:\ x_i \in X_i\ \forall i\right\}. \tag{5}$$

The Lagrangian subproblem for each i decouples Problem (5)

$$\{max_{x_i}\ f_i(x_i)-\lambda^{\mathrm{T}}(h_i(x_i)c\ );\ \text{subject to}:\ x_i \in X_i\}. \tag{6}$$

The dual variables are updated from the master dual problem as follows:

$$\left\{min_\lambda = \sum_i g_i(\lambda)\ +\lambda^{\mathbf{T}}\boldsymbol{c};\ \text{subject to}:\ \lambda\geqslant 0\right\}, \tag{7}$$

where $g_i(\lambda)$ is the dual function obtained as the maximum value of the Lagrangian solved in Problem (6) for a given λ. Thus, a gradient method can solve Problem (7), and the subgradient for each $g_i(\lambda)$ obtained by $s_i(\lambda) = -h_i(x_i^*(\lambda))$, where $_i(\lambda)$ is the optimal solution of Problem (6) for a given λ. The global subgradient is $s(\lambda) = \sum_i s_i(y) + c = c-\sum_i h(x_i^*(\lambda))$. Problem (6) can be independently and locally solved with knowledge of λ.

3 Objective function representation based on decomposition methods

3.1 Separable optimization

The choice of decomposition (of objective function) influences the choice of the algorithm for solving the corresponding mathematical program.

Definition 1: Separable optimization Problem (Horst et al., 2000)

$$\{min_{x\in\Re^n}F_0(x)\ \textit{subject to}: F_i(x)\leqslant b_i,\ l_i\leqslant x_i\leqslant u_i,\ i=1,...,m\}, \tag{8}$$

where $F_i(x) = \sum_{j=1}^n F_{ij}(x_j),\ i = 0,1,...,m.$

3.2 Factorable optimization

McCormick (1983, 1974, 1976) introduced factorable programming. A factorable program takes the following form

$$\{min_{x\in\Re^n}X^L(x)\ \textit{subject to}: l_i\leqslant X^i(x)\leqslant u_i,\ i=1,...,L-1\}, \tag{9}$$

where $X^i:\Re^n\mapsto\Re$

$X^i(x) = x_i$ for $i = 1,...,n$ and $X^p(x)$, $p = 1,...,i-1$, function X^i is $X^i(x) = \sum_{p=1}^{i-1} T_p^i(X^p(x)) + \sum_{p=1}^{i-1}\sum_{q=1}^{p} V_{q,p}^i (X^p(x)).U_{p,q}\ (X^q(x))$, where T's, U's, and V's are the transformation functions of a single variable. The lower and upper bounds $l_i\leqslant u_i$ are given constants. The function $X^i(x), i = 1,...,L$ can be written as factorable functions. McCormick (1974) developed a factorable programming language integrated with SUMT (Mylander et al., 1971) for NLPs. The functions $X^i(x),\ i = 1,...,L$ are called concomitant variable functions (cvfs). The cvfs includes separable and quadratic terms.

3.3 Almost block separable optimization

The following problem is considered:

$$min_{x\in R^n}f(x)\ = f_1(u,y) + f_2(v,y), \tag{10}$$

where $x = (u,v,y) \in \Re^n$ *and* $u \in \Re^{n1}, v \in \Re^{n2}, y \in \Re^{n3}$, $n_1 + n_2 + n_3 = n$, and y are called complicated variables [usually $n_1, n_2 \gg n_3$]

Let $\varphi_1(y)\ = min_u f_1(u,y), \varphi_2(y)\ = min_v f_2(v,y)$. The problem is equivalent to:

$$min_y\varphi_1(y)\ +\varphi_2(y). \tag{11}$$

If f_1 and f_2 are convex, then $\varphi_1(y)$ and $\varphi_2(y)$ are convex.

3.4 DC optimization problems

3.4.1 Continuous DC programming

One of the special non-convex programs is DC programming. DC function and dual DC programming are defined as follows:

Definition 2: DC function (Horst et al., 2000; Wu et al., 2018)

A real-valued function $f:\ \Re^n\mapsto\Re\ \cup\ \{+\infty,-\infty\}$ subject to:

$$\{f(x)\ = g(x)\ -h(x),\ \forall x \in \Re^n\}, \tag{12}$$

where $g,\ h:\Re^n\mapsto\Re\ \cup +\infty$ is a convex function and is a

DC function for any h and g.

Definition 3: DC program (Horst et al., 2000; Wu et al., 2018)

The following model is called a DC program

$$\{min\ f_0(x)\ \text{subject to}: f_i(x) \leqslant 0,\ \forall i = 1,2...,n\}, \quad (13)$$

if $f_i(x)$ are DC functions ($i = 0,1,2,...,n$) and it is the same as the following DC program. Then,

$$inf_{x_\in <\Re^n} f(x) \ = g(x) - h(x). \quad (14)$$

Hartman Theorem 1. The following DC programs are equal:

$$\begin{cases} sup\ f(x) : x \in C,\ f,\ C: \ \text{convex} \\ \inf\ g(x) - h(x) : x \in \Re^n,\ g,\ h: \ \text{convex} \\ \inf\ g(x) - h(x) : x \in C,\ f_1(x) - f_2(x) \\ \leqslant 0,\ g,\ h,\ f_1,\ f_2,\ C: \text{all convex} \end{cases}$$

Hartman Theorem 2. A function f is locally DC if an $\in$ -ball on which DC exists. Every function that is locally DC is considered a DC proposition. Let f_i be DC functions for $i = 1,...,m$. Thus, $\left\{\sum_i \lambda_i f_i(x) \text{ for } \lambda_i \in \Re\right\}$; $\{max_i f_i(x)\}$; $\{min_i f_i(x)\}$; $\{\Pi_i f_i(x)\}$; and $\{f_i\}$ are twice continuously differentiable DC. Moreover, (gof) is DC if f is DC and g is convex, and every continuous function on C (convex set) is the limit of a sequence of uniformly converging DC functions.

Definition 4: Subgradient of convex function (Horst et al., 2000; Wu et al., 2018)

A vector x^* is a subgradient of a convex function h at a point x if $h(z) \geqslant h(x) + \langle x^*, z - x\rangle$, where $\langle x,y\rangle = \sum_{i=1}^n x_i\ y_i$ is the inner product of two vectors with the same dimension. The subdifferential of $h(x)$ is the set of all subgradients.

Definition 5: Conjugate functions (Horst et al., 2000; Wu et al., 2018)

A conjugate function $h^*: \Re^n \mapsto \Re \cup +\infty$ of a convex function $h: \Re^n \mapsto \Re \cup +\infty$ is:

$$h^*(p) := sup_{y \in \Re^n} \langle y,x\rangle - h(x). \quad (15)$$

Theorem 3: The conjugate function $h^*(y)$ of $h(x)$ is convex. If $h(x)$ is a closed proper convex function, then the bi-conjugate of h is itself, that is, $h^{**} = h$.

Theorem 4 (Toland–Singer duality): Given closed convex functions $g,\ h: \Re^n \mapsto \Re \cup +\infty$, then:

$$inf_{x \in \Re^n}\{g(x) - h(x)\} = inf_{p \in \Re^n}\{h^*(p) - g^*(p)\}. \quad (16)$$

Definition 6: DC algorithm (Horst et al., 2000; Wu et al., 2018)

The following algorithm is used for obtaining a local optimal solution for the DC program.

Step 0 Find an inintial solution $x^0 \in dom_R(g)$. Set $t := 0$.

Step 1 Find $p^t \in \partial_R h(x^t)$.

Step 2 Find $p^{t+1} \in \partial_R g^*(p^t)$, where g^* is the "conjugate" of g.

Step 3 If $f(x^{t+1}) = f(x^t)$, stop. Otherwise, set $t := t + 1$, go to Step 1.

where $x^{t+1} = argmin_{y \in R^n}\{g(y) - h(x^t) - \langle p, y - x^t\rangle\}$.

3.4.2 Continuous relaxations for discrete DC programming

The positive support of $x \in Z^n$ is presented as follows: $supp^+(x) := \{i \in \{1,2,...,n\} : x_i > 0\}$.

The indicator vector χ_S is defined by:

$$\chi_S(i) = \begin{cases} 1 & i \in S \\ 0 & i \notin S \end{cases}.$$

$M^\natural$-convex and $L^\natural$-convex are two common discrete functions:

1) $M^\natural$-convex functions are defined as $\forall x,y \in \mathbb{Z}^n and\ i \in supp^+(x-y)$, function h: $\mathbb{Z}^n \mapsto \mathbb{Z} \cup +\infty$ is $M^\natural$-convex if it satisfies:

$$h(x) + h(y) \geqslant min\{h(x - \chi_i) + h(x + \chi_i)\}, \quad (17)$$

$$min_{j \in supp^+(x-y)} h(x - \chi_i + \chi_j) + h(y + \chi_i - \chi_j). \quad (18)$$

2) $L^\natural$-convex functions are defined as $\forall x,\ y \in \mathbb{Z}^n$, $h: \mathbb{Z}^n \mapsto \mathbb{Z} \cup +\infty$ is $L^\natural$ -convex if it satisfies:

$$h(x) + h(y) \geqslant h\left(\left\lceil \frac{x+y}{2} \right\rceil\right) + h\left(\left\lfloor \frac{x+y}{2} \right\rfloor\right). \quad (19)$$

Consider the following discrete DC program:

$$\{Inf\ f(x) = g(x) - h(x) \ \text{subject to}: x \in \mathbb{Z}^n\}. \quad (20)$$

The four kinds of discrete DC programs include $M^\natural - L^\natural$, $M^\natural - M^\natural$, $L^\natural - L^\natural$, and $L^\natural - M^\natural$, wherein the first three are NP-hard, and the last one on $\{0,1\}^n$ is in P, can be defined on the basis of $M^\natural$ and $L^\natural$-convex function definitions (Kobayashi, 2014; Maehara et al., 2018) .

We assume functions $g,h: \mathbb{Z}^n \mapsto \Re \cup \{+\infty\}$. The effective domain of g is $dom_Z g := \{x \in \mathbb{Z}^n : g(x) < +\infty\}$.

The convex closure $\overline{g}(x): \Re^n \mapsto \Re \cup \{+\infty\}$ of g is:

$$\overline{g}(x) = sup\{s(x) : s \text{ is an affine function},$$

$$s(y) \leqslant g(y)(y \in \mathbb{Z}^n)\}. \quad (21)$$

A convex extension $\hat{g}: \Re^n \mapsto \Re \cup \{+\infty\}$ of g is a convex function with the same function value on $x \in \mathbb{Z}^n$.

We assume

$\tilde{f}(x) := \overline{g}(x) - \hat{h}(x)$. Then $\tilde{f}(x) := g(x) - h(x),\ \forall x \in \mathbb{Z}^n$.

Thus:

$$inf_{x \in \mathbb{Z}^n}\{g(x) - h(x)\} = inf_{x \in \mathbb{Z}^n}\tilde{f}(x) \geqslant inf_{x \in \Re^n}\tilde{f}(x). \quad (22)$$

Theorem 5: For convex extensible functions g, h : $\mathbb{Z}^n \mapsto \Re \cup +\infty$ with $dom_Z g$ bounded and $dom_z g \in dom_Z h$:

$$inf_{z\in\mathbb{Z}^n}\{g(z)-h(z)\} = inf_{x\in\Re^n}\left\{\overline{g}(x)-\hat{h}(x)\right\}, \quad (23)$$

where $\overline{g}(x)$ is the linear closure of $g(x)$, and $\hat{h}(x)$ is any convex extension of $h(x)$.

We found that the discrete DC programming (20) is equivalent to the corresponding continuous relaxation DC programming based on Theorem 5.

3.5 DI optimization problems

Total and partial monotonicity are related to monotonicity for all and some variables with many GO applications. The d.i. monotonic optimization with increasing functions in $\Re_+^n$ can be generally described as follows:

$$\{min\ f(x)-g(x)\ \text{subject to}: f_i(x)-g_i(x)\leqslant 0,\ i=1,...,m\}. \quad (24)$$

Let $g(x) = 0$, and,

$$\forall i, f_i(x)-g_i(x)\leqslant 0 \Leftrightarrow max_{1\leqslant i\leqslant m}\{f_i(x)-g_i(x)\}\leqslant 0$$

$$\Leftrightarrow F(x)-G(x)\leqslant 0, \quad (25)$$

with increasing F, and G $\left(F(x)=max_i\{f_i(x)+\sum_{i\neq j}g_j(x)\},\ G(x)=\sum_i g_i(x)\right)$.

Then, the problem is reduced to:

$$\{minf(x);\ \text{subject to}:\ F(x)+t\leqslant F(b),\ G(x)+t\geqslant F(b),$$

$$0\leqslant t\leqslant F(b)-F(0),\ x\in[0,b]\subset\Re_+^n\}. \quad (26)$$

For any x, x' where $x'\leqslant x$, *if* $x\in G$, then $x'\subseteq G$, a set $G\subseteq\Re_+^n$ is normal.

Many GO problems, including polynomial, multiplicative, Lipschitz optimization problems, and non-convex quadratic programming, can be considered monotonic optimization problems.

3.6 Decomposition and multi-objective optimization

We consider the following problems:

$$P1: min_{x\in D\subseteq\Re^n}\ F(x)=f_1(x)+...+f_k(x), \quad (27)$$

$$P2: min_{x\in D\subseteq\Re^n}\ f(x)=(f_1(x),...,f_k(x)). \quad (28)$$

Objective function $F(x)$ in many GO problems can be represented by the summation of k relatively simple functions as $F(x) = f_1(x) + f_2(x) + ... + f_k(x)$. P2 is a multi-objective optimization problem. Let $E(f,D) \subseteq D$ be the set of all Pareto optimal solutions in D. We obtain the following theorems for optimal solutions of P1 and the optimal Pareto frontier of P2.

Theorem 6: If $\overline{x}$ is an optimal solution of P1, then $x\in E(f, D)$ of P2.

Theorem 7: Let $h_i(t)$ be a monotonic increasing function for $i = 1,...,k$. We consider the multi-objective optimization problem $min_{x\in D\subseteq\Re^n}h(x) = (h_1(f_1(x)),...,h_m(f_k(x)))$. Then, $E(f, D) = E(h, D)$. (Miettinen, 1999; Chinchuluun and Pardalos, 2007; Pardalos et al., 2008; Du and Pardalos, 2013; Migdalas et al., 2013; Pardalos et al., 2017)

Theorems 1 and 2 show that the extended Pareto optimal frontier set $E(h,D)$ can be obtained by solving P2 and searching for the optimal $\overline{x}$ of P1 from $E(h,D)$.

P2 can be a multi-objective optimization problem (MaOP). The algorithms for solving MaOPs can be classified as: 1) **Algorithm adaptation methods**, which modify/extend the classical EMO algorithms for solving MaOPs, including preference-based MOEA (PICEA; PBEA), Pareto-based MOEA (NSGA-II; SPEA2), indicator-based MOEA (HypE; SMSEMOA), decomposition-based MOEA (MOEA/D; M2M); and 2) **Problem transformation methods**, which transform the MaOP into a problem with few objectives, including objective selection (σ-MOSS; k-EMOSS; L-PCA) and objective extraction (Gu, 2016) . Refer to Gu (2016) and Mane and Rao (2017), for a review of solution algorithms and real-world applications of MaOPs, such as flight control system, engineering design, data mining, nurse scheduling, car controller optimization, and water supply portfolio planning.

MOEA/D is a mostly used method for solving P2. Its goals can be categorized as: 1) convergence to detect solutions close to the Pareto frontier; 2) diversity to determine well-distributed solutions; and 3) coverage to cover the entire Pareto frontier. Several MOEAs for these goals are found in literature, which can be broadly categorized under three categories, namely, 1) domination-, 2) indicator-, and 3) decomposition-based frameworks (Ehrgott and Gandibleux, 2000; Trivedi et al., 2017).

In MOEA/D literature, three decomposition methods, including the weighted sum (WS), the weighted Tchebycheff (TCH), and penalty based boundary intersection (PBI) approaches.

The ith subproblem of the WS approach is given as:

$$min\ g^{ws}(x|\lambda_i)=\sum_{j=1}^{m}\lambda_i^j f_j(x). \quad (29)$$

This method is efficient for solving convex Pareto solutions with min objective function.

The ith subproblem of the TCH approach is defined as follows:

$$min\ g^{te}(x|\lambda_i,z^*)=max_{1\leqslant j\leqslant m}\{\lambda_i^j|f_j(x)-z_j^*|\}, \quad (30)$$

where $z^* = (z_1^*,...,z_m^*)^{\mathrm{T}}$ is the ideal reference point with $z_j^* < min\{f_j(x)|x \in \Omega\}$ for $j = 1,2,...,m$.

The ith subproblem of the PBI approach is defined as follows:

$$min\ g^{pbi}(x|\lambda_i,z^*) = d_1 + \theta d_2, \tag{31}$$

where $d_1 = \frac{// ((F(x)-z^*)^{\mathrm{T}}\lambda_i //}{// \lambda_i //}$ and $d_2 = \left\| F(x) - \left(z^* - d_1 \frac{\lambda_i}{// \lambda_i //} \right) \right\|$. z^* is the reference point shown in (32), and θ is a penalty parameter that should be tuned properly.

4 Kolmogorov's superposition

Kolmogorov (1956) presents the following theorem as Kolmogorov's superposition:

Theorem 8: Continuous real functions $\psi^{p,q}(x)$ (for any integer $n \geqslant 2$) on the closed unit interval $E^1 = [0,1]$ exists similar to continuous real function $f(x_1,...,x_n)$ on the n-dimensional unit cube E^n, which can be shown as:

$$f(x_1,...,x_n) = \sum_{q=1}^{2n+1} \chi_q \left[\sum_{p=1}^{n} \psi^{pq}(x_p) \right], \tag{32}$$

$$y^q = \sum_{p=1}^{n} \psi^{pq}(x_p), \tag{33}$$

where $\chi_q(y)$ is a continuous real function (refer to (Arnol'd, 1959; Tikhomirov, 1991) for a brief proof of the theorem). The following equation is obtained for $n = 3$, by setting, $\varphi_q(x_1, x_2) = \psi^{1q}(x_1) + \psi^{2q}(x_2)$ and $h_q(y, x_3) = \chi_q y + \psi^{3q}(x_3)$: $f(x_1, x_2, x_3) = \sum_{q=1}^{7} [\varphi_q(x_1, x_2), x_3]$.

The application of Kolmogorov theorem in GO in space-filling curve is an example of its efficient optimizing functions based on their projection from n dimensions to one dimension (Goertzel, 1999; Lera and Sergeyev, 2010; Sergeyev et al., 2013). Sprecher (Sprecher and Draghici, 2002; Sprecher, 2013; Sprecher, 2014) explored the link between the aforementioned theorem and the space-filling curves from computational algorithms for real-valued continuous functions.

5 Conclusions

This paper reviewed different GO and decomposition methods on the basis of objective function representation. Many GO methods are derived from the branch and bound method, which are inefficient for finding a remarkable solution. This paper provides opportunity for additional research on decomposition techniques based on objective function representation, multi-objective optimization, and Kolmogorov's superposition. The development of other parallel decomposition-based GO methods based on the objective function representation for MINLP, such as Decogo solver (Nowak et al., 2018), can be a challenging area in MINLP solver development. Kolmogorov theorem in GO will be discussed in future studies.

Industry 4.0 is known as the future of smart manufacturing and industrial revolution. Making decentralized decision is critical in Industry 4.0 (Marques et al., 2017). Horizontal and vertical integrations are two principal characteristics in Industry 4.0. Decentralized decision support systems are needed depending on the different types of decisions, including operational, tactical, real-time, and strategic. Many optimization problems are integrated with artificial intelligence in Industry 4.0, in which decision makers (DMs) should make a decentralized decision. This paper will help DMs in Industry 4.0 represent their objective function based on different GO techniques, such as Kolmogorov's superposition and DC programming, which can be solved separately. Finally, Khakifirooz, Pardalos, et al. (2018) and Khakifirooz, Chien, et al. (2018) reported that applications of non-convex optimization in decision support system development for smart manufacturing and Industry 4.0 can be a challenging direction for future research.

Acknowledgements Professor Pardalos' research is partially supported by the Paul and Heidi Brown Preeminent Professorship at ISE, University of Florida. Dr. Mahdi Fathi would like to thank Prof. Murray Brown and Mrs. Helen Brown for their encouragement and support during this research.

References

Alperin H, Nowak I (2005). Lagrangian smoothing heuristics for max-cut. Journal of Heuristics, 11(5–6): 447–463

Arnol'd V I (1959). On the representation of continuous functions of three variables by superpositions of continuous functions of two variables. Matematicheskii Sbornik, 90(1): 3–74

Ausiello G, Crescenzi P, Gambosi G, Kann V, Marchetti-Spaccamela A, Protasi M (2012). Complexity and Approximation: Combinatorial Optimization Problems and Their Approximability Properties. Berlin: Springer Science & Business Media

Babayev D A, Bell G I (2001). An optimization problem with a separable non-convex objective function and a linear constraint. Journal of Heuristics, 7(2): 169–184

Becker R W, Lago G (1970). A global optimization algorithm. In: Proceedings of the 8th Allerton Conference on Circuits and Systems Theory. 3–12

Bertsekas D P (1999). Nonlinear Programming. Belmont: Athena Scientific

Bliek C (1998). Coconut deliverable d1-algorithms for solving nonlinear and constrained optimization problems. The COCONUT Project

Boddy M S, Johnson D P (2002). A new method for the global solution of large systems of continuous constraints. In: Bliek C, Jermann C,

Neumaier A, eds. International Workshop on Global Optimization and Constraint Satisfaction. Berlin: Springer

Boender C G E, Romeijn H E (1995). Stochastic methods. In: Pardalos P M, Romeijin H E, eds. Handbook of global optimization. Berlin: Springer

Bomze I M, Csendes T, Horst R, Pardalos P M (1997). Developments in Global Optimization. Berlin: Springer Science & Business Media

Boyd S, Xiao L, Mutapcic A, Mattingley J (2007). Notes on Decomposition Methods. Stanford: Stanford University

Burkard R E, Kocher M, Rüdolf R (1997). Rounding strategies for mixed integer programs arising from chemical production planning. Yugoslav Journal of Operations Research

Chiang M, Low S H, Calderbank A R, Doyle J C (2007). Layering as optimization decomposition: A mathematical theory of network architectures. Proceedings of the IEEE, 95(1): 255–312

Chinchuluun A, Pardalos P M (2007). A survey of recent developments in multiobjective optimization. Annals of Operations Research, 154 (1): 29–50

Dantzig G B, Wolfe P (1960). Decomposition principle for linear programs. Operations Research, 8(1): 101–111

Dixon L C W, Szegö G P (1974). Towards global optimisation. In: Proceedings of a workshop at the University of Cagliari, Italy

Du D Z, Pardalos P M (2013). Handbook of Combinatorial Optimization: Supplement, Vol. 1. Berlin: Springer Science & Business Media

Duran M A, Grossmann I E (1986). An outer-approximation algorithm for a class of mixed-integer nonlinear programs. Mathematical Programming, 36(3): 307–339

Ehrgott M, Gandibleux X (2000). A survey and annotated bibliography of multiobjective combinatorial optimization. OR-Spektrum, 22(4): 425–460

Fisher M L (1980). Worst-case analysis of heuristic algorithms. Management Science, 26(1): 1–17

Fletcher R, Leyffer S (1994). Solving mixed integer nonlinear programs by outer approximation. Mathematical Programming, 66(1–3): 327–349

Floudas C, Aggarwal A, Ciric A (1989). Global optimum search for nonconvex NLP and MINLP problems. Computers & Chemical Engineering, 13(10): 1117–1132

Floudas C A (2013). Deterministic Global Optimization: Theory, Methods and Applications, Vol. 37. Berlin: Springer Science & Business Media

Floudas C A, Pardalos P M (2013). State of the Art in Global Optimization: Computational Methods and Applications, Vol. 7. Berlin: Springer Science & Business Media

Floudas C A, Pardalos P M (2014). Recent Advances in Global Optimization. Princeton: Princeton University Press

Forrest S (1993). Genetic algorithms: Principles of natural selection applied to computation. Science, 261(5123): 872–878

Geoffrion A M (1972). Generalized benders decomposition. Journal of Optimization Theory and Applications, 10(4): 237–260

Glover F, Laguna M (1997). Tabu Search. Berlin: Springer

Goemans M X, Williamson D P (1995). Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. Journal of the Association for Computing Machinery, 42(6): 1115–1145

Goertzel B (1999). Global optimization with space-filling curves. Applied Mathematics Letters, 12(8): 133–135

Grossmann I E (2002). Review of nonlinear mixed-integer and disjunctive programming techniques. Optimization and Engineering, 3(3): 227–252

Grossmann I E, Kravanja Z (1997). Mixed-integer nonlinear programming: A survey of algorithms and applications. In: Biegler L T, Colleman T F, Conn A R, Samtosa F N, eds. Large-scale Optimization with Applications. Berlin: Springer

Gu F Q (2016). Many objective optimization: Objective reduction and weight design. Dissertation for the Doctoral Degree. HongKong: HKBU

Henrion D, Lasserre J B (2002). Solving global optimization problems over polynomials with gloptipoly 2.1. In: Proceedings of International Workshop on Global Optimization and Constraint Satisfaction. Berlin: Springer

Hirsch M J, Meneses C, Pardalos P M, Resende M G (2007). Global optimization by continuous grasp. Optimization Letters, 1(2): 201–212

Hochbaum D, Jansen K, Rolim J D, Sinclair A (1999). Randomization, Approximation, and Combinatorial Optimization. Algorithms and Techniques: In: Proceedings of 3rd International Workshop on Randomization and Approximation Techniques in Computer Science, and 2nd International Workshop on Approximation Algorithms for Combinatorial Optimization Problems. Berlin: Springer Science & Business Media

Holmberg K (1990). On the convergence of cross decomposition. Mathematical Programming, 47(1–3): 269–296

Holmberg K, Ling J (1997). A Lagrangian heuristic for the facility location problem with staircase costs. European Journal of Operational Research, 97(1): 63–74

Hooker J (2011). Logic-Based Methods for Optimization: Combining Optimization and Constraint Satisfaction, Vol. 2. Hoboken: John Wiley & Sons

Horst R, Pardalos P M (2013). Handbook of Global Optimization, Vol. 2. Berlin: Springer Science & Business Media

Horst R, Pardalos P M, Van Thoai N (2000). Introduction to Global Optimization. Berlin: Springer Science & Business Media

Horst R, Tuy H (2013). Global Optimization: Deterministic Approaches. Berlin: Springer Science & Business Media

Kelly F P, Maulloo A K, Tan D K (1998). Rate control for communication networks: Shadow prices, proportional fairness and stability. Journal of the Operational Research Society, 49(3): 237–252

Kesavan P, Allgor R J, Gatzke E P, Barton P I (2004). Outer approximation algorithms for separable nonconvex mixed-integer nonlinear programs. Mathematical Programming, 100(3): 517–535

Khakifirooz M, Chien C-F, Pardalos F M, Panos M (2018). Management Suggestions on Semiconductor Manufacturing Engineering: An Operations Research and Data Science Perspective. Berlin: Springer

Khakifirooz M, Pardalos P M, Fathi M, Power D J (2018). Decision support for smart manufacturing. Encyclopedia of IST, 5th ed, IGI Global Book

Kirkpatrick S, Gelatt C D Jr, Vecchi M P (1983). Optimization by simulated annealing. Science, 220(4598): 671–680

Kobayashi Y (2014). The complexity of maximizing the difference of two matroid rank functions, METR2014–42. University of Tokyo

Kocis G R, Grossmann I E (1987). Relaxation strategy for the structural optimization of process flow sheets. Industrial & Engineering Chemistry Research, 26(9): 1869–1880

Kojima M, Kim S, Waki H (2003). A general framework for convex relaxation of polynomial optimization problems over cones. Journal of the Operations Research Society of Japan, 46(2): 125–144

Kolmogorov A (1956). On the Representation of Continuous Functions of Several Variables as Superpositions of Functions of Smaller Number of Variables. Berlin: Springer

Lasserre J B (2001). Global optimization with polynomials and the problem of moments. SIAM Journal on Optimization, 11(3): 796–817

Lera D, Sergeyev Y D (2010). Lipschitz and Hölder global optimization using space-filling curves. Applied Numerical Mathematics, 60(1–2): 115–129

Li D, Sun X, Wang J, McKinnon K I (2009). Convergent Lagrangian and domain cut method for nonlinear knapsack problems. Computational Optimization and Applications, 42(1): 67–104

Locatelli M (2002). Simulated annealing algorithms for continuous global optimization. In: Horst R, Pardalos P M, eds. Handbook of Global Optimization. Berlin: Springer

Maehara T, Marumo N, Murota K (2018). Continuous relaxation for discrete DC programming. Mathematical Programming, 169(1): 199–219

Mane S U, Rao M N (2017). Many-objective optimization: Problems and evolutionary algorithms–A short review. International Journal of Applied Engineering Research, 12(20): 9774–9793

Marques M, Agostinho C, Zacharewicz G, Jardim-Goncalves R (2017). Decentralized decision support for intelligent manufacturing in industry 4.0. Journal of Ambient Intelligence and Smart Environments, 9(3): 299–313

Mart R, Panos P, Resende M (2018). Handbook of Heuristics. Berlin: Springer

Mawengkang H, Murtagh B (1986). Solving nonlinear integer programs with large-scale optimization software. Annals of Operations Research, 5(2): 425–437

McCormick G P (1974). A mini-manual for Use of the Sumt Computer Program and the Factorable Programming Language. Stanford: Stanford University

McCormick G P (1976). Computability of global solutions to factorable nonconvex programs: Part iconvex underestimating problems. Mathematical Programming, 10(1): 147–175

McCormick G P (1983). Nonlinear Programming: Theory, Algorithms, and Applications. New York: Wiley

Metropolis N, Rosenbluth A W, Rosenbluth M N, Teller A H, Teller E (1953). Equation of state calculations by fast computing machines. Journal of Chemical Physics, 21(6): 1087–1092

Miettinen K (1999). Nonlinear Multiobjective Optimization. International Series in Operations Research and Management Science. Berlin: Springer

Migdalas A, Pardalos P M, Värbrand P (2013). Multilevel Optimization: Algorithms and Applications, Vol. 20. Berlin: Springer Science & Business Media

Mockus J (2012). Bayesian Approach to Global Optimization: Theory and Applications, Vol. 37. Berlin: Springer Science & Business Media

Moré J J, Wu Z (1997). Global continuation for distance geometry problems. SIAM Journal on Optimization, 7(3): 814–836

Mylander W C, Holmes R L, McCormick G P (1971). A guide to sumt-version 4: The computer program implementing the sequential unconstrained minimization technique for nonlinear programming (Technical Report RAC-P-63). Mclean: Research Analysis Corporation

Neumaier A (2004). Complete search in continuous global optimization and constraint satisfaction. Acta Numerica, 13: 271–369

Nowak I (2005). Relaxation and decomposition methods for mixed integer nonlinear programming, Vol. 152. Berlin: Springer Science & Business Media

Nowak I, Breitfeld N, Hendrix E M, Njacheun-Njanzoua G (2018). Decomposition-based inner-and outerrefinement algorithms for global optimization. Journal of Global Optimization, (4–5): 1–17

Nowak M P, Römisch W (2000). Stochastic lagrangian relaxation applied to power scheduling in a hydrothermal system under uncertainty. Annals of Operations Research, 100(1–4): 251–272

Padberg M, Rinaldi G (1991). A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems. SIAM Review, 33(1): 60–100

Palacios-Gomez F, Lasdon L, Engquist M (1982). Nonlinear optimization by successive linear programming. Management Science, 28 (10): 1106–1120

Palomar D P, Chiang M (2006). A tutorial on decomposition methods for network utility maximization. IEEE Journal on Selected Areas in Communications, 24(8): 1439–1451

Pardalos P M (1991). Global optimization algorithms for linearly constrained indefinite quadratic problems. Computers & Mathematics with Applications (Oxford, England), 21(6–7): 87–97

Pardalos P M, Migdalas A, Pitsoulis L (2008). Pareto optimality, game theory and equilibria, Vol. 17. Berlin: Springer Science & Business Media

Pardalos P M, Rosen J B (1986). Methods for global concave minimization: A bibliographic survey. SIAM Review, 28(3): 367–379

Pardalos P M, Rosen J B (1987). Constrained Global Optimization: Algorithms and Applications. New York: Springer-Verlag

Pardalos P M, Wolkowicz H (1998). Topics in semidefinite and interior-point methods. American Mathematical Society

Pardalos P M, Zilinskas A, Zilinskas J (2017). Non-convex multi-objective optimization, Vol. 123. Berlin: Springer

Paules G E I V IV, Floudas C A (1989). Apros: Algorithmic development methodology for discrete-continuous optimization problems. Operations Research, 37(6): 902–915

Pintér J D (1996). Global Optimization in Action. Dordrecht: Kluwer Academic Publishers

Rahmaniani R, Crainic T G, Gendreau M, Rei W (2017). The benders decomposition algorithm: A literature review. European Journal of Operational Research, 259(3): 801–817

Resende M G C, Ribeiro C C (2003). Greedy randomized adaptive search procedures. In: Glover F, Kochenberger G, eds. Hand Book of Metaheuristics. Dordrecht: Kluwer Academic Publishers

Rockafellar R T (2016). Problem decomposition in block-separable convex optimization: Ideas old and new. In: Proceedings of the 5th Asian Conference on Nonlinear Analysis and Optimization, Niigata,

Japan

Sahinidis N V (1996). Baron: A general purpose global optimization software package. Journal of Global Optimization, 8(2): 201–205

Schelstraete S, Schepens W, Verschelde H (1999). Energy minimization by smoothing techniques: A survey. Molecular Dynamics: from Classical to Quantum Methods

Schichl H (2010). Mathematical Modeling and Global Optimization. Cambridge: Cambridge University Press

Sellmann M, Fahle T (2003). Constraint programming based Lagrangian relaxation for the automatic recording problem. Annals of Operations Research, 118(1–4): 17–33

Sergeyev Y D, Strongin R G, Lera D (2013). Introduction to Global Optimization Exploiting Space-Filling Curves. Berlin: Springer Science & Business Media

Smith E M, Pantelides C C (1996). Global optimisation of general process models. In: Grossmann I E, eds. Global Optimization in Engineering Design. Berlin: Springer

Smith E M, Pantelides C C (1999). A symbolic reformulation/spatial branch-and-bound algorithm for the global optimisation of non-convex minlps. Computers & Chemical Engineering, 23(4–5): 457–478

Sprecher D (2013). Kolmogorov superpositions: A new computational algorithm. In: Igelnik B, eds. Efficiency and Scalability Methods for Computational Intellect. New York: IGI Global

Sprecher D (2014). On computational algorithms for real-valued continuous functions of several variables. Neural Networks, 59: 16–22

Sprecher D A, Draghici S (2002). Space-filling curves and Kolmogorov superposition-based neural networks. Neural Networks, 15(1): 57–67

Strongin R, Sergeyev Y D (2000). Global Optimization with Non-Convex Constraints. Dordrecht: Kluwer Academic Publishers

Svanberg K (2002). A class of globally convergent optimization methods based on conservative convex separable approximations. SIAM Journal on Optimization, 12(2): 555–573

Tawarmalani M, Sahinidis N V (2002). Convexification and Global Optimization in Continuous and Mixedinteger Nonlinear Programming: Theory, Algorithms, Software, and Applications, Vol. 65. Berlin: Springer Science & Business Media

Tikhomirov V (1991). On the representation of continuous functions of several variables as superpositions of continuous functions of one variable and addition. In: Kolmogorov A N, Shiryaeu A, eds. Selected Works of AN Kolmogorov. Berlin: Springer

Torn A, Zilinskas A (1989). Global Optimization. New York: Springer-Verlag

Trivedi A, Srinivasan D, Sanyal K, Ghosh A (2017). A survey of multiobjective evolutionary algorithms based on decomposition. IEEE Transactions on Evolutionary Computation, 21(3): 440–462

Türkay M, Grossmann I E (1996). Logic-based minlp algorithms for the optimal synthesis of process networks. Computers & Chemical Engineering, 20(8): 959–978

Vaidyanathan R, El-Halwagi M (1996). Global optimization of nonconvex minlps by interval analysis. In: Grossmann I E, eds. Global Optimization in Engineering Design. Berlin: Springer

Van Hentenryck P, Michel L, Deville Y (1997). Numerica: A Modeling Language for Global Optimization. Boston: MIT Press

Vazirani V V (2013). Approximation Algorithms. Berlin: Springer Science & Business Media

Vecchietti A, Grossmann I E (1999). Logmip: A disjunctive 0–1 non-linear optimizer for process system models. Computers & Chemical Engineering, 23(4–5): 555–565

Viswanathan J, Grossmann I E (1990). A combined penalty function and outer-approximation method for minlp optimization. Computers & Chemical Engineering, 14(7): 769–782

Westerlund T, Lundqvist K (2001). Alpha-ECP, version 5.01: An interactive MINLP-solver based on the extended cutting plane method

Westerlund T, Pettersson F (1995). An extended cutting plane method for solving convex minlp problems. Computers & Chemical Engineering, 19: 131–136

Westerlund T, Pettersson F, Grossmann I E (1994). Optimization of pump configurations as a minlp problem. Computers & Chemical Engineering, 18(9): 845–858

Wu C, Wang Y, Lu Z, Pardalos P M, Xu D, Zhang Z, Du D Z (2018). Solving the degree-concentrated fault-tolerant spanning subgraph problem by DC programming. Mathematical Programming, 169(1): 255–275

Zamora J M, Grossmann I E (1998a). A global minlp optimization algorithm for the synthesis of heat exchanger networks with no stream splits. Computers & Chemical Engineering, 22(3): 367–384

Zamora J M, Grossmann I E (1998b). Continuous global optimization of structured process systems models. Computers & Chemical Engineering, 22(12): 1749–1770

Zhang H, Wang S (2006). Global optimization of separable objective functions on convex polyhedra via piecewise-linear approximation. Journal of Computational and Applied Mathematics, 197(1): 212–217

Zheng Q P, Wang J, Pardalos P M, Guan Y (2013). A decomposition approach to the two-stage stochastic unit commitment problem. Annals of Operations Research, 210(1): 387–410

Zwick U (1999). Outward rotations: A tool for rounding solutions of semidefinite programming relaxations, with applications to max cut and other problems. In: Proceedings of the 31st Annual ACM Symposium on Theory of Computing, ACM, 679–687

Applied Soft Computing Journal 73 (2018) 383–393

Contents lists available at ScienceDirect

Applied Soft Computing Journal

journal homepage: www.elsevier.com/locate/asoc

A multi-objective evolutionary algorithm based on decomposition and constraint programming for the multi-objective team orienteering problem with time windows

Wanzhe Hu [a,b], Mahdi Fathi [b,*], Panos M. Pardalos [b]

[a] College of Materials Science and Engineering, Chongqing University, Chongqing, China
[b] Department of Industrial and Systems Engineering, Center for Applied Optimization, University of Florida, Gainesville, USA

HIGHLIGHTS

- We study the multi-objective team orienteering problem with time windows (MOTOPTW).
- A multi-objective evolutionary algorithm based on decomposition and constraint programming (CPMOEA/D) is developed.
- Using constraint programming as an improvement approach is promising for solving the MOTOPTW.
- Compared with reported results on benchmark instances, many new non-dominated objective vectors are found.

ARTICLE INFO

Article history:
Received 2 May 2018
Received in revised form 5 August 2018
Accepted 21 August 2018
Available online xxxx

Keywords:
Multi-objective combinatorial optimization
Team orienteering problem
Multi-objective evolutionary algorithm
Decomposition approach
Constraint programming

ABSTRACT

The team orienteering problem with time windows (TOPTW) is a well-known variant of the orienteering problem (OP) originated from the sports game of orienteering. Since the TOPTW has many applications in the real world such as disaster relief routing and home fuel delivery, it has been studied extensively. In the classical TOPTW, only one profit is associated with each checkpoint while in many practical applications each checkpoint can be evaluated from different aspects, which results in multiple profits. In this study, the multi-objective team orienteering problem with time windows (MOTOPTW), where checkpoints with multiple profits are considered, is introduced to find the set of Pareto optimal solutions to support decision making. Moreover, a multi-objective evolutionary algorithm based on decomposition and constraint programming (CPMOEA/D) is developed to solve the MOTOPTW. The advantages of decomposition approaches to handle multi-objective optimization problems and those of the constraint programming to deal with combinatorial optimization problems have been integrated in CPMOEA/D. Finally, the proposed algorithm is applied to solve public benchmark instances. The results are compared with the best-known solutions from the literature and show more improvement.

1. Introduction

The orienteering problem (OP) is an NP-hard combinatorial optimization problem introduced by Tsiligirides [1]. The OP can be defined as follows. Given one specified control point and a set of checkpoints, the travel times between any two points are known and each checkpoint is associated with a profit. The goal of the OP is to determine a tour for an individual who starts at the control point, tries to visit as many checkpoints as possible, and returns to the control point within a given time limit. Its objective is to maximize the total profits collected by visiting checkpoints. The OP is also known as the maximum collection problem [2], the selective traveling salesman problem [3], and the bank robber problem [4]. The survey regarding the traveling salesman problems with profits [5] clearly describes the differences between the OP and other routing problems. Recent reviews on the OP, its variants, and applications are presented in [6,7].

The team orienteering problem with time windows (TOPTW) is an extension of the OP. The OP determines only one tour while the TOPTW determines more than one tours and takes time window constraints into consideration. There are many TOPTW applications in real life, for example, the tourist trip design problem (TTDP) [8], athlete recruitment from high schools [2], and routing technicians to service customers [9]. Consider the TTDP as an example. It is often impossible for tourists to visit all places of interest in a region during one or more days. Hence, they have to make feasible routes to visit more valuable attractions within the limited time. If we describe each attraction as a checkpoint with a profit representing its attractiveness to tourists, this route planning problem for multiple day trips can be formulated as a

* Corresponding author.
E-mail address: mahdi.fathi@ufl.edu (M. Fathi).

https://doi.org/10.1016/j.asoc.2018.08.026
1568-4946/

TOPTW. Its objective is to maximize the total profits collected by visiting selected attractions. Many researchers have focused their attention on the TTDP, the TOPTW or related variants [10–13] since Vansteenwegen and Van Oudheusden presented their model on the mobile tourist guide [8]. However, in real-world applications, each checkpoint can have several different kinds of profits. For instance, in the TTDP one attraction may have different attractivenesses to different tourists. Then the objective becomes to maximize a vector of profits instead of a scalar. Therefore, a multi-objective optimization approach to provide decision makers with the solutions considering different kinds of profits is needed. This motivates us to study the multi-objective TOPTW (MOTOPTW).

During the last decade, several solution methods have been proposed for the single-objective TOPTW. Two exact approaches were reported. The branch and price algorithm introduced by [14] is the first exact algorithm to deal with TOPTW. Among 117 benchmark instances, they can solve 91 instances to optimality within two-hour time limit. Another exact method is the constraint programming (CP) approach presented by [15]. The 117 instances mentioned above are solved using CP and the results show that CP is a more competitive solution method. On the other hand, a large number of metaheuristics are reported. Several representative metaheuristics are based on the ant colony system algorithm [16,17], the iterated local search heuristic [18,19], the variable neighborhood search heuristic [20], the simulated annealing heuristic [21], the granular variable neighborhood search algorithm [22], the iterative three-component heuristic [23], and the artificial bee colony algorithm [24]. These publications have demonstrated that approaches based on metaheuristics are efficient and competitive for the single-objective TOPTW. However, it is clear that they cannot be applied directly to solve the MOTOPTW.

There are few studies on the multi-objective OP or its variants including the MOTOPTW in the literature. Schilde et al. [25] proposed two multi-objective optimization algorithms, the Pareto ant colony optimization algorithm (P-ACO) and the Pareto variable neighborhood search algorithm (P-VNS), to solve the bi-objective OP. A multi-objective artificial bee colony algorithm was reported recently [26] and the experimental results on benchmarks show that it is a more efficient approach than P-ACO and P-VNS. However, these algorithms are only able to solve the OP without the time windows constraints. Chen et al. [27] developed a multi-objective ant colony optimization algorithm to solve the OP with time window constraints. Nevertheless, its solutions contain only one tour and cannot be applied to the MOTOPTW. Actually, the approaches applied to the MOTOPTW where each checkpoint is associated with several profits were not found in the literature although several multi-objective algorithms for related routing problems with multiple tours were proposed [28,29]. Therefore, in this study we introduce the MOTOPTW and propose a competitive multi-objective evolutionary algorithm to solve it.

The main contributions of this paper are as follows: (1) the MOTOPTW is proposed, where several different kinds of profits can be associated with each checkpoint; (2) a multi-objective evolutionary algorithm based on decomposition and constraint programming is developed to solve the MOTOPTW; (3) Public benchmark instances are solved using the developed algorithm, and compared with reported results, many new non-dominated objective vectors are found.

The remainder of the paper is organized as follows. In Section 2, we discuss the problem description and formulation. Section 3 provides the proposed solution method. Section 4 presents the numerical results and discussion on benchmark instances. Finally, we conclude the paper and present future research directions in Section 5.

2. Problem description and formulation

Given one control point and N checkpoints, each of which is associated with multiple profits, a service time, and a time window, the goal of MOTOPTW is to determine K routes such that different kinds of profits collected by visiting each checkpoint no more than once are maximized. Note that individuals must visit each checkpoint within the predefined time window, and one has to wait if arriving at one checkpoint before its start time window. In addition, the total time spent by one individual is limited. Specifically, the MOTOPTW is modeled with a graph $G = (V \cup \{d\}, E)$, where $V = \{1, 2, \ldots, N\}$ is the vertex set representing the checkpoint set, $d = 0$ represents the control point which can be visited more than once, and $E = \{(i, j)|i, j \in V\}$ is the set of edges. Other notations used for the model are given as follows.

Parameters

K: the number of routes to be generated

st_i: the service time of the checkpoint $i \in V$ and $st_0 = 0$

p_i^k: the kth profit of the checkpoint i where $i \in V$, and $k \in \{1, \ldots, m\}$

t_{ij}: the time required to travel from a checkpoint i to another checkpoint j, $i, j \in V$ and $i \neq j$

T_{max}: the time limit for each individual

$[b_i, c_i]$: the time window for each checkpoint $i \in V$

M: a large constant

Decision Variables

x_{ij}: a binary variable equal to 1 if one checkpoint $j \in V$ is visited right after another one $i \in V$ and 0 otherwise.

u_i: a continuous variable representing the start time to visit the checkpoint $i \in V$

The MOTOPTW mathematical formulation:

$$\text{Maximize } F(x) = \left(\sum_{i=0}^{N}\sum_{j=0}^{N} p_i^1 \cdot x_{ij}, \sum_{i=0}^{N}\sum_{j=0}^{N} p_i^2 \cdot x_{ij}, \ldots, \sum_{i=0}^{N}\sum_{j=0}^{N} p_i^m \cdot x_{ij}\right)^T \quad (1)$$

s.t.

$$\sum_{j=1}^{N} x_{0j} = \sum_{j=1}^{N} x_{j0} = K \quad (2)$$

$$\sum_{j=0}^{N} x_{ij} = \sum_{j=0}^{N} x_{ji} \leq 1, \forall i = 1, \ldots, N \quad (3)$$

$$u_i + t_{ij} + st_i - u_j \leq M(1 - x_{ij}), \quad \forall i = 0, \ldots, N, \forall j = 1, \ldots, N, i \neq j \quad (4)$$

$$u_i + t_{i0} + st_i \leq T_{max}, \forall i = 1, \ldots, N \quad (5)$$

$$b_i \leq u_i \leq c_i, \forall i = 1, \ldots, N \quad (6)$$

$$x_{ij} \in \{0, 1\}, \forall i, j = 0, \ldots, N, i \neq j \quad (7)$$

$$u_i \geq 0, \forall i = 0, \ldots, N \quad (8)$$

The objective function (1) is to maximize a vector composed of m profits collected by all K routes. Constraints (2) ensure that the control point is visited K times and then K routes will be generated. Constraints (3) guarantee that each checkpoint can be visited at most once. Constraints (4) state the time connectivity between checkpoints visited by each individual, and eliminate possible subtours. Constraints (5) guarantee that the total time spent by each individual cannot exceed the maximum time limit. Constraints (6) are the time window constraints. Constraints (7) and (8) define the decision variables.

3. Solution method

Using an exact algorithm such as branch and bound to solve a multi-objective combinatorial optimization problem is time-consuming, particularly for large-scale instances. Consequently, specialized metaheuristics have become prevalent in recent research [30]. In this study, we propose a constraint programming (CP) and decomposition based multi-objective evolutionary algorithm (CPMOEA/D) to deal with the MOTOPTW.

3.1. Multi-objective optimization

Before describing the solution method in detail, some definitions regarding multi-objective optimization are presented [31]. For $F : \Omega \rightarrow R^m$, a multi-objective optimization problem (MOP) can be represented as follows:

$$\text{Maximize } F(x) = (f_1(x), \ldots, f_m(x))^T$$

$$\text{Subject to } x \in \Omega \tag{9}$$

where x, Ω, and m are the decision variable, the decision space, and the number of conflicting objectives, respectively. $F : \Omega \rightarrow R^m$ is composed of m objective functions and R^m is the objective space. In an MOP, an objective vector v is said to dominate another one u if and only if $v_i \geq u_i, i \in 1, \ldots, m$ holds with at least one strict inequality. An objective vector is non-dominated if no other objective vectors dominate it, and a solution x is said to be Pareto optimal if its objective vector is non-dominated by others. The set of non-dominated objective vectors is called the Pareto front (PF) and the set of Pareto optimal solutions constitute the Pareto set (PS). Since it is generally time-consuming to obtain a complete PF, in real-life applications an approximation to the PF is required to support decision-making.

3.2. Multi-objective evolutionary algorithm based on decomposition

The multi-objective evolutionary algorithm (MOEA) is very popular in solving MOPs because it is able to obtain multiple Pareto-optimal solutions in a single simulation run. The three goals of an MOEA are: (1) to find a set of objective vectors as close as possible to the PF (convergence); (2) to find a set of well distributed objective vectors (diversity); and (3) to cover the entire PF (coverage). Although there are several different kinds of MOEAs developed for these goals, decomposition-based algorithms have attracted the most attention of researchers since the decomposition based multi-objective evolutionary algorithm (MOEA/D) is introduced by Zhang and Li [32]. In the framework of MOEA/D, an MOP is decomposed into multiple scalar optimization subproblems based on the decomposition approach. These subproblems are associated with different weight vectors. At each generation, a population of solutions is maintained to store the best solution found so far for each subproblem. The main feature of MOEA/D is that the neighborhood relations among these subproblems are given based on the distance between their weight vectors. They assure that the optimal solutions of neighboring subproblems should be similar and any information from a neighboring subproblem could be helpful for optimizing one subproblem. Therefore, evolutionary operators can be applied to two neighboring solutions to produce better solutions. These subproblems are optimized simultaneously by evolving this solution population. Several studies have been reported for solving multi-objective routing problems with MOEA/D based approaches which perform well [33,34]. Hence, the MOEA/D framework is adopted in our algorithm.

In the literature, there are two popular decomposition methods, namely, the weighted sum approach and the Tchebycheff approach (TCH). In the weighted sum approach, the ith subproblem is defined as $Min\ g^{ws}(x|\lambda_i) = \sum_{j=1}^{m} \lambda_i^j f_j(x)$. This approach works efficiently for multi-objective linear programming problems but it cannot approximate the entire PF for multi-objective combinatorial optimization problems [35]. Therefore, the TCH is adopted in this study and the definition is given in the next section.

3.3. Main procedures of CPMOEA/D

One of the main advantages of the MOEA/D is its ability to integrate with some scalar optimization methods such as the CP and problem-specific heuristics, since each subproblem is a scalar optimization problem. CP has been proved to be an efficient method for combinatorial optimization problems, especially scheduling problems [36]. Thus, it is natural to integrate CP with MOEA/D to solve multi-objective combinatorial optimization problems. The proposed algorithm for the MOTOPTW, CPMOEA/D, is an attempt for this kind of integration. The primary procedure of CPMOEA/D is given as follows.

The TCH is employed in CPMOEA/D to decompose the MOTOPTW into N scalar optimization subproblems. The objective of the jth subproblem is:

$$\min_{x\in\Omega} g^{te}(x|\lambda^j, z^*) = \min_{x\in\Omega} \max_{1\leq i\leq m} \{\lambda_i^j |f_i(x) - z_i^*|\} \tag{10}$$

where Ω , $\lambda^j = (\lambda_1^j, \ldots, \lambda_m^j)^T$, and $z^* = (z_1^*, \ldots, z_m^*)^T$ are the decision space, the weight vector of the jth subproblem, and the reference point, respectively.

The CPMOEA/D works as follows:

Input:

- The MOTOPTW;
- *MaxIteration*: the max number of iterations;
- N: the number of the subproblems in CPMOEA/D;
- $\{\lambda^1, \ldots, \lambda^N\}$: a collection of uniformly distributed weight vectors;
- T: the size of the neighborhood of each subproblem;
- p_c: the probability of crossover;
- p_m: the probability of mutation.

Output:

- *EP*: an external population to store non-dominated solutions found.

Step 1. Initialization:

1.1. Build the CP model for the subproblems of MOTOPTW where the objectives are associated with weight vectors of corresponding subproblems and the current reference point according to the TCH;

1.2. Determine the neighborhood of each subproblem based on the Euclidean distances of weight vectors between any two subproblems. Set $B(i) = \{i_1, \ldots, i_T\}$ for each subproblem $i = 1, \ldots, N$. The set of $i_1, \ldots, i_T$ is the indexes of T closest subproblems;

1.3. Generate initial solutions $\{x^1, \ldots, x^N\}$ for all subproblems randomly, and set $FV^i = F(x^i)$;

1.4. Set $z^* = (0, \ldots, 0)$ and $EP = \varnothing$.

Step 2. Update: For $i = 1, \ldots, N$, do

2.1. Select two indexes k, l from $B(i)$ randomly;

2.2. Apply the crossover operator with probability p_c and the mutation operator with probability p_m on solutions x_k and x_l to generate a new solution y;

2.3. Set the objective function of the CP model with the weight vector of the ith subproblem and current reference point. The CP solver is created based on the CP model to improve the solution y to the solution y';

2.4. For each objective $j = 1, \ldots, m$, if $z_j > f_j(y')$, set $z_j = f_j(y')$;

2.5. For each neighboring subproblem $j \in B(i)$, if $g^{te}(y'|\lambda^j, z^*) \leq g^{te}(x^j|\lambda^j, z^*)$, set $x^j = y'$ and $FV^j = F(y')$;

2.6. If no vectors in EP dominate $F(y')$, then add $F(y')$ to EP and remove all the vectors in EP dominated by $F(y')$.

Step 3. Stopping Criteria: If the stopping criteria is satisfied, then stop and output EP. Otherwise, go to **Step 2**.

Note that in initialization, the closest subproblem of each subproblem is itself, and the ith subproblem is included in $B(i)$ which is the place where two parents are located. The reference point z is an important parameter, which is closely related to the approximation of the PF and the distribution of non-dominated vectors. Different from the classical MOEA/D where the reference point is updated whenever one new solution is generated, in this algorithm, there are two subprocesses where the reference point is fixed and many new solutions can be found. This means that it is possible that new objective vectors obtained may be better than the reference point at one objective. As a result, the absolute representation used in the original form of Eq. (10) will probably result in missing good solutions. To avoid this kind of possibility, improved objectives of the subproblems are proposed below in the two subprocesses of CPMOEA/D.

3.4. Solution representation

Solving TOPTW involves three kinds of decisions: selecting the checkpoints to visit, partitioning the checkpoints into groups, and sequencing the checkpoints in each group to get several tours. In CPMOEA/D, an indirect encoding method is adopted. One solution is encoded as an individual, a sequence π of all checkpoints usually called a giant tour, to determine the visiting sequence. A subsequence of π denoted by $(i, l_i)_\pi$ where i is the index of its starting point and l_i is the number of checkpoints following i in π is a tour if visiting this subsequence from the control point is feasible. A feasible solution of a giant tour is a set of its tours without shared checkpoints, and obviously many feasible solutions are available. To get the optimal solution, a decoding procedure called "Optimal Split" is required to select and partition these checkpoints. The "Optimal Split" procedure dedicated to the team orienteering problem was first proposed by [37].

The basic idea of the "Optimal Split" procedure is given as follows. First, the pool of tours considered in this procedure has to be determined, and then a set of tours is selected from the pool so that the objective is minimized. Since the number of tours increases dramatically with the problem size n, it is time-consuming to enumerate all tours. Therefore, the term, saturated tour, is introduced to reduce the number of the considered tours. A tour is said to be saturated if the corresponding subsequence cannot be extended anymore. That is, the checkpoint $\pi[i + l_i]$ is the last one in the sequence or the subsequence $(i, l_i + 1)_\pi$ will result in an infeasible tour. Actually, the best solution can be obtained by considering only saturated tours. In addition, the same condition holds for the subproblems of MOTOPTW, and the proof is presented in the following.

Proposition 1. *If an optimal solution S of a giant tour contains K tours which are not all saturated and unvisited checkpoints are available, another optimal solution S' with K all saturated tours can be found.*

Proof. Although multiple objectives are involved, one optimal objective vector v can be determined with respect to the objective of one subproblem. Assume that K tours denoted by $(i^1, l_{i^1}), \ldots, (i^K, l_{i^K})$ are an optimal solution of a sequence π. Two tours (i^k, l_{i^k}) and $(i^{k+1}, l_{i^{k+1}})$ are said to be adjacent if the Equation $\pi(i^k + l_{i^k} + 1) = \pi(i^{k+1})$ holds. If a tour has no adjacent tours, such a tour must be saturated. Otherwise, one or more checkpoints following this tour can be included to get another objective vector dominating v, which means better subproblem objective value and conflicts with the assumption. If a tour (i^k, l_{i^k}) has an adjacent tour and is not saturated, we can extend it by repetitively adding the first checkpoint in its adjacent tour until this tour becomes saturated. As the solution is optimal, the checkpoint making the tour (i^k, l_{i^k}) saturated must precede the checkpoint $\pi(i^{k+1} + l_{i^{k+1}})$. Otherwise, the tour used to visit $(i^{k+1}, l_{i^{k+1}})$ is able to visit more checkpoints and collect more profits, which conflicts with the assumption. It is obvious that such an extending procedure can be used repetitively to make unsaturated tours of an optimal solution saturated while the objective vector remains unchanged.

According to the research conducted by [38], selecting an optimal set of tours from a pool of saturated tours can be viewed as a knapsack problem with interval conflict graphs. However, multiple objectives are considered in our study, and the algorithms used in previous research cannot work anymore. As a result, based on the work by [39], an algorithm specific to our subproblems in CPMOEA/D is proposed in the next section.

3.5. The multi-objective knapsack problem with interval conflict graphs

First, the definition of an interval graph is given. A graph $G = (V, E)$ is an interval graph if each vertex $v \in V$ is associated with an interval of the real line denoted by $I_v = (a_v, b_v)$ with $a_v, b_v \in \mathbb{R}$ and $a_v < b_v$. Two vertices u, v are adjacent and an edge $e_{uv} \in E$ exists if and only if $I_u \cap I_v \neq \varnothing$. Then, let us consider a kind of interval conflict graphs. Given an interval graph $G = (V, E)$, assume that the vertices are indexed in non-decreasing order of b_i in related intervals $I_v = (a_i, b_i)$, that is, $i < j$ if $b_i \leq b_j$. Two vertices u, v are said to be in conflict if they are adjacent, i.e., $e_{uv} \in E$ exists. A special vertex with the index $prev_i$ can be identified such that for any two vertices $i, j \in V$, if $1 \leq j \leq prev_i$, vertices i, j are not in conflict, and if $prev_i < j < i$, they are in conflict. Finally, the multi-objective knapsack problem with interval conflict graphs (MOKPICG) is defined as follows. Given a knapsack with a limited volume and a set of items, each of which is associated with a volume and multiple kinds of values. Different from general knapsack problems, any two items may be in conflict with each other, which means they cannot be in the knapsack together. If we associate each item with an interval, the conflict relationship between these items can be described with an interval conflict graph. The goal of MOKPICG is to find a subset of non-dominated objective vectors so that the total volume of selected items does not conflict with each other and exceed the knapsack volume.

Considering each saturated tour as an interval in an interval conflict graph, in CPMOEA/D one subproblem to select an optimal set of K tours can be regarded as an MOKPICG with a decomposition based objective. The volume of the knapsack is K and the volume of each tour is 1. The objective for each subproblem is to minimize Eq. (10). It is related to the reference point and multiple objectives. To solve it, a dynamic programming (DP) algorithm is presented

below, which integrates the ideas from [38] and [39]. Three notations are involved in the DP. A vector $f = (f_1, f_2, \ldots, f_m, f_{m+1})$ is introduced to represent one solution. The first m elements are the values of corresponding m objectives and f_{m+1} denotes the volume of all involved items. C^i is the ith set to store solution vectors. D_r is a dominance relation between solution vectors. Specifically, f^1 is said to dominate f^2 if $f_i^1 \geq f_i^2, i \in \{1, \ldots, m\}$ and $f_{m+1}^1 \leq f_{m+1}^2$ holds. In addition, since the reference point is fixed in the process of DP, a solution with at least one objective better than that of the reference point is possible. To avoid the risk of losing better solutions, the objectives have been modified. Before presenting the algorithm in detail, some notations are defined.

V: the volume of the knapsack;
n: the number of items;
m: the number of objectives;
$v_i, i \in \{1, \ldots, n\}$: volumes of items;
$p_i^j, i \in \{1, \ldots, n\}$: the jth profit of the item i;
$prev_i, i \in \{1, \ldots, n\}$: as defined above;
$w_j, j \in \{1, \ldots, m\}$: the value of the weight vector for the jth objective;
$z_j, j \in \{1, \ldots, m\}$: the value of the jth element of the reference point.

Algorithm 1 DP

Input
V; n; m; v_i; p_i^j; $prev_i$; w_j; z_j;
Output
d^n: the best objective value of the subproblem;

$C^0 \leftarrow \{(0, \ldots, 0)\}$;
for $i \leftarrow 1$ to n **do**
  $C^{i1} \leftarrow C^{i-1}$;
  $C^{i2} \leftarrow \{f_1^{prev_i} + p_1^i, \ldots, f_m^{prev_i} + p_m^i | f_{m+1}^{prev_i} + v^i \leq V, f^{prev_i} \in C^{prev_i}\}$;
  * Assume that $C^{i1} = \{f^{i1(1)}, \ldots, f^{i1(n1)}\}$,$C^{i2} = \{f^{i2(1)}, \ldots, f^{i2(n2)}\}$ *\
  for $j \leftarrow 1$ to $n1$ **do**
    $dominates \leftarrow false$;
    $k \leftarrow 1$
    while $k \leq n2$ **do**
      if $f^{i2(k)} D_r f^{i1(j)}$ **then**
        $C^{i1} \leftarrow C^{i1} / \{f^{i1(j)}\}$
        break;
      else if $f^{i1(j)} D_r f^{i2(k)}$ **then**
        $C^{i2} \leftarrow C^{i2} / \{f^{i2(k)}\}$
        $dominates \leftarrow true$
        break;
      end if
      $k \leftarrow k + 1$
    end while
    if dominates **then**
      while $k \leq n2$ **do**
        if $f^{i1(j)} D_r f^{i2(k)}$ **then**
          $C^{i2} \leftarrow C^{i2} / \{f^{i2(k)}\}$
        end if
      end while
    end if
  end for
  $C^i \leftarrow C^{i1} \cup C^{i2}$;
end for
$d^n = inf$;
* Assume that $C^k = \{f^{k(1)}, \ldots, f^{k(l)}\}$ *\
for $i \leftarrow 1$ to l **do**
  $d' = inf$
  for $j \leftarrow 1$ to m **do**
    if $w_j = 1$ **then**
      $d' = z_j - v_j^{k(i)}$
      break;
    else if $j = m$ **then**
      $d' = \max_{1 \leq j \leq m} \{w_j * (z_j - v_j^{k(i)})\}$
    end if
  end for
  if $d^n > d'$ **then**
    $d^n \leftarrow d'$;
  end if
end for

3.6. CP improvement

The CP has been proved to be an efficient method for the TOPTW [15]. Based on the IBM ILOG CP Optimizer, a commercial CP solver, The CP model for TOPTW has been presented, and the numerical results show that it is comparable with state-of-the-art algorithms. Also, compared with most metaheuristics, the solving strategies used by CP is not stochastic, and the solving performance is stable and robust. However, the CP Optimizer at present only supports lexicographic multi-objective optimization but does not support Pareto optimization. Fortunately, each subproblem of MOEA/D is a scalar optimization problem. Therefore, it is natural to integrate CP with MOEA/D to solve MOTOPTW.

There are two main differences between CP models of our subproblems and that of a general TOPTW. One is that the objective of a subproblem is not to maximize one kind of profits but to minimize the objective function (10). Hence, it is associated with the reference point and weight vector. Similar to the MOKPICG, it is possible for CP Optimizer to find a solution non-dominated by the reference point. To avoid missing this kind of solutions, the objectives used in our CP model are as follows. If any scalar of the weight vector is equal to 1, say $w_j = 1$, then the objective is

$$\min_{x \in \Omega} \{z_j - f_j(x)\}. \tag{11}$$

Otherwise, the objective is

$$\min_{x \in \Omega} \max_{1 \leq j \leq m} \{\lambda_j * (z_j - f_j(x))\}. \tag{12}$$

Another difference is that an initial solution obtained by genetic operators is provided for the model. While finding a feasible solution for CP Optimizer is easy, it can take quite some times to improve it to a better one. The warm start capabilities of CP Optimizer are employed in CPMOEA/D by giving a good starting point solution that it will try to improve. As the initial solution becomes increasingly better, CP Optimizer can produce better solutions more quickly.

4. Numerical results

To confirm the effectiveness of the CP based MOEA/D for solving the MOTOPTW, computational experiments were conducted on a set of popular benchmark instances. We first describe the benchmark instances and then present the results and discussions.

4.1. Benchmark instances

76 benchmark instances are available from http://www.mech.kuleuven.be/en/cib/op and they are introduced by [16,40] and [18], respectively. Among them, 56 instances were converted from the instances introduced by [41] for the vehicle routing problem with time windows. To be specifically, six sets of points are involved, that is, c101–c109, c201–c208, r101–r112, r201–r211, rc101–rc108, rc201–rc208. Each of these instances contains 100 points and corresponding position coordinates, time windows, visiting time, and profits for all points are provided. In the instances

of c101–c109 and c201–c208, points are clustered while in r101–r112 and r201–r211 points are located remotely. Remote and clustered points are mixed in the instances of rc101–rc108 and rc201–rc208. In addition, points in the instances of c201–c208, r201–r211, and rc201–rc208 have longer time windows. Here, we focus our attention on these 56 instances so as to make this article concise and short. Note that, the travel times are not given directly. In our experiments, the travel times between any two points are equal to their corresponding Euclidean distances which are rounded down to the first decimal. In addition, all these instances originally have only one kind of profits, and to compare the results we obtained with existing results, we adopt the method given by [27] to extend them to bi-objective ones by setting p_{i1} to $p_{(i+1)2}$, which means the second profit of the point $i+1$ is equal to the first profit of the point i.

4.2. Performance analysis of CPMOEA/D

Although CP has been proved to be an efficient method to tackle the TOPTW, there are few publications reporting the integration of MOEA/D and CP. To demonstrate the effectiveness of the integration, the results obtained by CPMOEA/D are compared with that given by three other algorithms, namely, the multi-objective ant colony algorithm proposed by [27] (MOACO), a multi-objective algorithm developed by replacing the CP process of CPMOEA/D with the iterated local search heuristic proposed by [18] as the improvement algorithm (IMOEA/D), and another multi-objective algorithm based on CP and TCH (MOCP) where no evolutionary operators are included. The details of MOCP are presented in the following.

MOCP works as follows:

Input:

- The MOTOPTW;
- *MaxIteration*: the max number of iterations;
- N: the number of the subproblems in MOCP;
- $\{\lambda^1, \ldots, \lambda^N\}$: a collection of uniformly distributed weight vectors;

Output:

- EP: an external population to store found non-dominated solutions.

Step 1. Initialization:

1.1. Build the CP model for the subproblems of MOTOPTW where the objectives are associated with weight vectors of corresponding subproblems and the current reference point according to TCH;

1.2. Set $z^* = (0, \ldots, 0)$ and $EP = \varnothing$.

Step 2. Update: For $i = 1, \ldots, N$, do

2.1. Set the objective function of the CP model with the weight vector of the ith subproblem and current reference point. The CP solver is created based on the CP model to improve solution x^i to solution y;

2.2. For each objective $j = 1, \ldots, m$, if $z_j > f_j(y)$, set $z_j = f_j(y)$;

2.3. If $g^{te}(y|\lambda^i, z^*) \leq g^{te}(x^i|\lambda^i, z^*)$, set $x^i = y$ and $FV^i = F(y)$;

2.4. If no vectors in EP dominate $F(y)$, then add $F(y)$ to EP and remove all the vectors in EP dominated by $F(y)$.

Step 3. Stopping Criteria: If the stopping criteria is satisfied, then stop and output EP. Otherwise, go to **Step 2**.

To analysis the performance of the CPMOEA/D and MOCP, the following popular performance metrics are used [42].

Set coverage (SC): Given two PFs, P and Q, $SC(P, Q)$ is defined as the percentage of the solutions in Q dominated by at least one solution in P. That is,

$$SC(P, Q) = \frac{|\{q \in Q | \exists p \in P : p \text{ dominates } q\}|}{|Q|} \tag{13}$$

where $|X|$ is the number of the components in X. Considering P as the true PF and Q as an approximation, this metric is to evaluate the closeness of Q to P. The lower the value of $SC(P, Q)$ is, the better solutions in Q we have.

Numbers of non-dominated objective vectors (NS): This metric gives the number of non-dominated objective vectors of a set. It is defined as

$$NS(P, P^*) = |\{v | v \in P \wedge v \in P^*\}| \tag{14}$$

where $|X|$ is the number of the components in X, and P^* is the PF. The bigger the NS is, the better non-dominated set we have.

Spacing (SP): This metric is introduced to measure the distribution of the non-dominated set obtained in the objective space. It is defined as

$$SP = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(d_i - \bar{d})^2} \tag{15}$$

where n is the number of the non-dominated set Q, $d_i = \min_{k \in Q \wedge k \neq i} \sum_{m=1}^{M} |f_m^i - f_m^k|$, and $\bar{d} = \sum_{i=1}^{n} d_i/n$. The lower the value of SP, the better non-dominated set we have.

Inverted generational distance (IGD): The IGD can be defined by the uniformly distributed true PF P^* and an approximation to it P:

$$IGD(P, P^*) = \sum_{v \in P} \frac{d(v, P^*)}{|P^*|} \tag{16}$$

where $d(v, P^*)$ is the minimum Euclidean distance between v and all points in P^*. This metric considers both the convergence and diversity of the solutions, and to get a better IGD, the set P is required to be close to P^*. The smaller the IGD is, the better non-dominated set we obtain.

Since the true PF is unknown for all instances, it seems that solutions which are non-dominated by solutions provided by other algorithms could make their way into the true PF. The algorithm CPMOEA/D was implemented in Python and the IBM ILOG CP optimizer 12.8.0 was used as the constraint programming solver. All experiments were run on a computer with Intel(R) Core(TM) i5-3210M 2.5 GHz and 8.00 GB of RAM. The parameters associated with the MOEA/D were set as follows: $MaxIteration = 150$, $N = 30$, and $T = 3$. The weight vectors were generated in the same way with MOEA/D in [32] where N is controlled by H, a positive integer. Since each solution was coded as a permutation of all checkpoints, the cycle crossover, and exchange mutation operators for the traveling salesman problem [43] were used to produce new offsprings. The crossover rate (p_c) and the mutation rate (p_m) were set to 0.9 and 0.3, respectively. Note that using CP to improve one solution is time-consuming. To obtain the good results in reasonable time, some limitations have to be made to control the time spent by CP Optimizer. Based on experiment analysis, two kinds of limitations were adopted. First, the CP improvement step (step 2.3 in CPMOEA/D) was called every 50 iterations. That is, only when the number of iterations is 0, 50, or 100, CP optimizer is invoked. Second, the CP model terminated with the best feasible solution at the end of 100,000 times failures (the parameter *FailLimit* in CP optimizer). To guarantee the computational resources consumed by CP optimizer in CPMOEA/D and MOCP were the same, the maximum iterations in MOCP was set to 3. In IMOEA/D, the parameters associated with MOEA/D were

Table 1
Comparisons of *SC* between MOACO, MOCP, IMOEA/D, and CPMOEA/D.

Instance	MOACO	MOCP	IMOEA/D	CPMOEA/D	Instance	MOACO	MOCP	IMOEA/D	CPMOEA/D
c101	0.22	0.00	0.33	0.00	c201	0.82	0.58	1.00	0.13
c102	0.75	0.25	1.00	0.00	c202	1.00	0.78	1.00	0.23
c103	0.80	0.70	0.92	0.08	c203	0.93	0.90	0.56	0.08
c104	0.54	0.83	0.60	0.15	c204	0.79	0.93	1.00	0.18
c105	0.20	0.18	0.82	0.08	c205	1.00	0.64	0.93	0.14
c106	0.33	0.20	0.55	0.00	c206	1.00	1.00	0.92	0.07
c107	0.75	0.55	0.36	0.00	c207	1.00	0.50	1.00	0.08
c108	1.00	0.36	0.64	0.00	c208	0.81	0.67	0.71	0.39
c109	0.40	0.15	0.85	0.00					
r101	0.33	0.00	0.00	0.00	r201	1.00	0.46	1.00	0.24
r102	0.56	0.00	0.56	0.00	r202	1.00	1.00	0.00	1.00
r103	0.42	0.15	0.50	0.10	r203	1.00	1.00	0.19	0.50
r104	0.81	0.25	0.88	0.09	r204	1.00	1.00	0.21	0.75
r105	0.71	0.08	0.57	0.00	r205	1.00	1.00	0.44	0.33
r106	0.11	0.11	0.40	0.00	r206	1.00	0.50	0.68	0.40
r107	0.58	0.53	0.63	0.10	r207	1.00	1.00	0.30	0.60
r108	1.00	0.33	0.92	0.00	r208	1.00	0.80	0.40	0.00
r109	0.68	0.27	0.95	0.11	r209	1.00	1.00	0.41	0.14
r110	0.55	0.88	0.21	0.48	r210	1.00	1.00	0.21	0.67
r111	0.47	0.67	0.88	0.30	r211	0.84	1.00	0.08	0.90
r112	0.80	0.88	0.62	0.22					
rc101	0.57	0.00	0.27	0.00	rc201	1.00	1.00	0.63	0.67
rc102	0.00	0.60	0.00	0.00	rc202	1.00	1.00	0.89	0.00
rc103	0.90	0.86	0.33	0.50	rc203	0.80	0.63	0.38	0.33
rc104	0.90	0.86	0.33	0.10	rc204	0.89	0.67	0.53	0.00
rc105	0.63	0.50	0.00	0.00	rc205	0.90	1.00	0.71	0.13
rc106	0.57	0.20	0.00	0.00	rc206	1.00	1.00	0.20	0.20
rc107	0.60	0.36	0.38	0.08	rc207	1.00	1.00	0.31	0.67
rc108	0.92	0.30	0.67	0.00	rc208	1.00	1.00	0.00	1.00

Table 2
Comparisons of *NS* between MOACO, MOCP, IMOEA/D, and CPMOEA/D.

Instance	MOACO	MOCP	IMOEA/D	CPMOEA/D	Instance	MOACO	MOCP	IMOEA/D	CPMOEA/D
c101	7	8	6	9	c201	2	5	0	13
c102	3	9	0	14	c202	0	2	0	10
c103	2	3	1	11	c203	1	1	8	11
c104	6	2	4	11	c204	4	1	0	14
c105	8	9	2	11	c205	0	4	1	12
c106	6	8	5	10	c206	0	0	1	14
c107	3	5	7	11	c207	0	5	0	12
c108	0	7	4	13	c208	3	5	5	11
c109	6	11	2	13					
r101	6	10	10	10	r201	0	7	0	13
r102	4	6	4	8	r202	0	0	18	0
r103	7	11	7	18	r203	0	0	13	6
r104	4	6	2	21	r204	0	0	15	2
r105	4	12	6	15	r205	0	0	14	6
r106	8	8	6	14	r206	0	1	8	6
r107	5	7	6	19	r207	0	0	19	4
r108	0	6	1	21	r208	0	2	12	4
r109	7	11	1	17	r209	0	0	13	6
r110	5	1	15	12	r210	0	0	19	2
r111	8	4	2	14	r211	3	0	22	1
r112	3	1	5	14					
rc101	3	12	8	12	rc201	0	0	3	5
rc102	4	2	5	5	rc202	0	0	2	11
rc103	1	1	4	6	rc203	3	3	8	10
rc104	1	1	8	9	rc204	2	2	8	2
rc105	3	4	6	6	rc205	1	0	5	7
rc106	3	8	6	11	rc206	0	0	16	8
rc107	6	7	8	11	rc207	0	0	11	4
rc108	1	7	5	13	rc208	0	0	21	0

set to the same values with CPMOEA/D. The pseudo code for the iterated local search heuristic was given in [18] and related parameters were set as follows: the start position in a tour to remove checkpoints S was initialized as a random number between 1 and the number of checkpoints in the solution under consideration, the number of the consecutive visits to remove R was initialized as 1, and *NumberofTimesNoImprovement* was set to 5. Note that, S and R become increasingly greater in the iteration process. Once they become infeasible for the shake step, they have to be reinitialized.

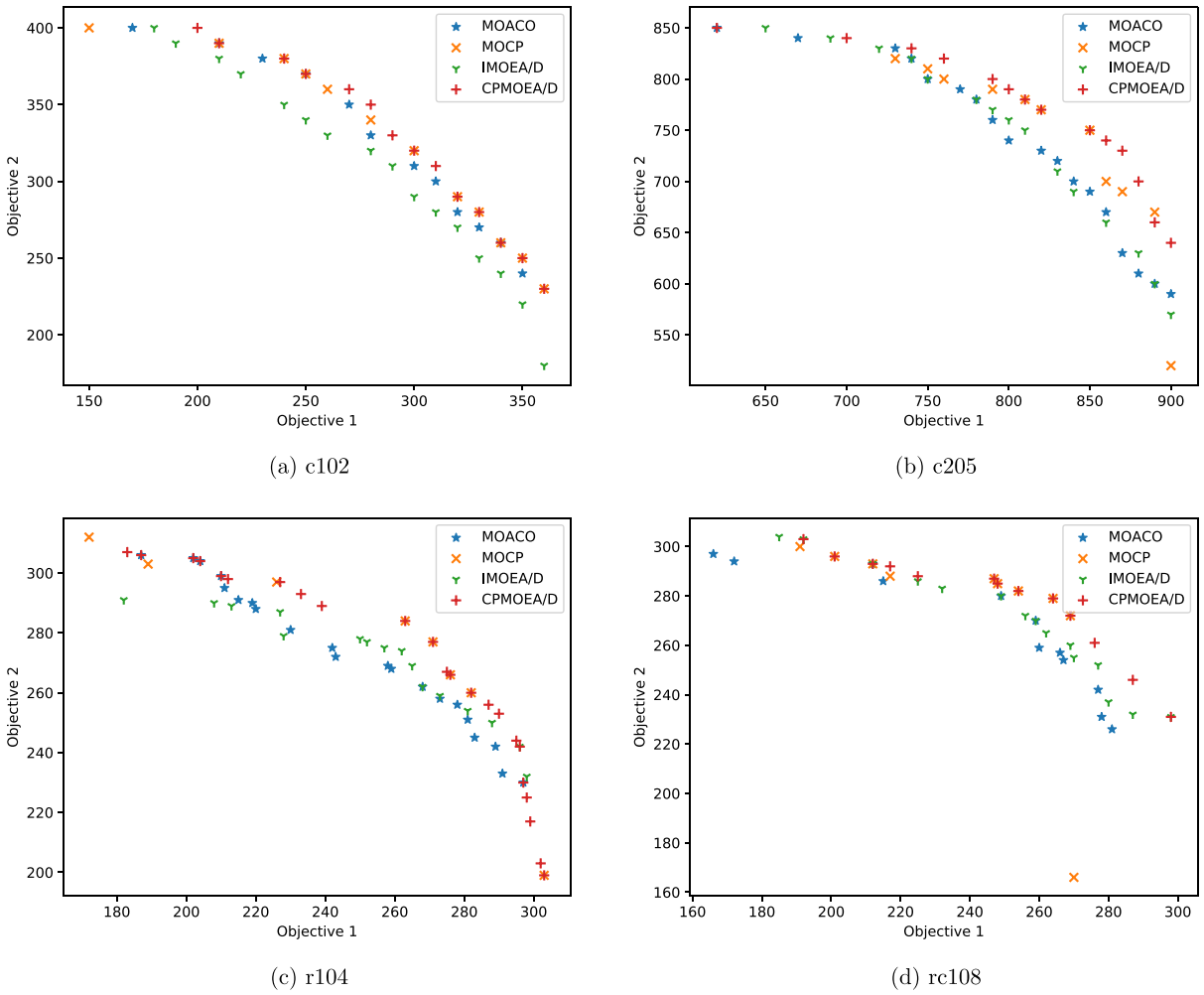
Table 3
Comparisons of *SP* between MOACO, MOCP, IMOEA/D, and CPMOEA/D.

Instance	MOACO	MOCP	IMOEA/D	CPMOEA/D	Instance	MOACO	MOCP	IMOEA/D	CPMOEA/D
c101	8.75	12.18	9.94	12.86	c201	10.29	26.81	9.00	50.73
c102	8.29	14.72	7.50	2.58	c202	12.75	39.00	18.77	29.23
c103	20.59	10.44	14.04	19.93	c203	25.27	14.46	10.83	40.69
c104	10.26	6.40	9.43	7.46	c204	7.52	30.61	9.33	12.94
c105	9.80	11.13	10.83	13.20	c205	12.39	39.42	9.98	18.46
c106	11.55	14.70	14.66	17.35	c206	22.55	16.39	9.48	45.68
c107	11.15	18.59	14.77	13.79	c207	16.29	50.16	18.20	30.52
c108	4.71	6.56	6.56	5.76	c208	21.20	30.91	6.44	17.95
c109	13.56	3.61	6.06	3.61					
r101	9.01	12.74	12.74	12.74	r201	37.87	8.64	10.79	7.82
r102	8.88	3.34	9.41	6.17	r202	15.10	8.41	6.93	26.69
r103	7.71	5.01	11.34	2.78	r203	6.02	10.14	8.85	12.68
r104	4.02	22.34	5.37	3.61	r204	9.21	11.69	6.03	13.36
r105	3.95	13.78	3.92	3.34	r205	8.73	21.09	8.78	6.72
r106	5.83	15.98	4.67	4.64	r206	14.84	0.00	5.68	14.03
r107	4.50	4.99	9.70	2.54	r207	5.84	13.05	5.75	22.70
r108	3.56	20.01	20.37	2.80	r208	12.05	20.14	10.28	23.67
r109	3.66	6.14	5.31	6.34	r209	36.53	38.44	4.60	0.64
r110	5.65	19.36	3.48	3.41	r210	21.17	16.52	10.42	33.48
r111	9.91	8.19	6.83	3.88	r211	10.43	28.11	11.06	5.36
r112	4.68	13.69	9.10	5.16					
rc101	5.26	2.96	3.89	2.96	rc201	8.40	5.90	11.56	11.75
rc102	9.53	4.79	8.52	8.52	rc202	7.29	0.00	9.13	24.93
rc103	8.27	14.38	4.06	1.44	rc203	4.99	31.63	14.84	14.87
rc104	8.46	32.86	11.58	8.87	rc204	5.96	43.37	8.32	0.00
rc105	4.82	7.18	7.09	7.09	rc205	10.79	0.00	7.48	11.43
rc106	9.30	12.18	12.41	8.74	rc206	6.97	29.23	5.82	16.50
rc107	5.72	6.63	6.17	5.73	rc207	8.14	0.00	7.59	17.20
rc108	3.99	29.44	3.92	7.27	rc208	16.07	28.07	12.48	11.64

Table 4
Comparisons of *IGD* between MOACO, MOCP, IMOEA/D, and CPMOEA/D.

Instance	MOACO	MOCP	IMOEA/D	CPMOEA/D	Instance	MOACO	MOCP	IMOEA/D	CPMOEA/D
c101	2.22	0.00	3.33	0.00	c201	8.27	11.42	41.31	1.77
c102	7.86	5.00	22.46	0.00	c202	38.51	19.39	58.54	5.00
c103	6.79	8.65	12.04	0.77	c203	24.41	12.85	28.94	4.71
c104	9.54	10.37	6.79	1.54	c204	15.93	12.82	54.80	2.32
c105	1.67	1.67	11.38	0.83	c205	38.32	15.17	28.62	2.86
c106	5.16	2.00	8.06	0.00	c206	31.31	11.81	26.87	8.67
c107	12.40	5.35	3.33	0.00	c207	48.13	7.75	36.56	1.60
c108	15.80	3.40	7.29	0.00	c208	13.85	15.46	24.10	7.70
c109	4.93	1.54	11.59	0.00					
r101	2.83	0.00	0.00	0.00	r201	30.22	7.15	33.87	5.13
r102	3.59	0.00	21.71	0.00	r202	28.09	18.55	0.00	12.02
r103	1.00	0.41	2.44	0.50	r203	55.26	9.10	1.03	13.07
r104	7.12	0.20	6.12	0.40	r204	93.38	44.10	4.01	13.22
r105	4.37	2.73	4.88	0.00	r205	40.73	10.33	9.43	5.42
r106	0.46	1.27	5.27	0.00	r206	44.97	0.93	25.38	10.43
r107	1.26	1.79	2.86	0.23	r207	43.63	13.97	3.40	6.79
r108	6.95	0.49	6.47	0.00	r208	67.28	12.41	29.28	0.00
r109	5.03	0.58	3.57	0.78	r209	20.47	12.41	12.31	0.26
r110	1.37	2.95	0.76	1.50	r210	29.72	40.44	2.68	6.72
r111	5.42	4.68	15.99	1.59	r211	13.17	11.46	0.48	6.15
r112	4.30	4.46	4.08	2.06					
rc101	2.83	0.00	1.40	0.00	rc201	122.68	89.99	33.43	48.06
rc102	0.00	11.46	0.00	0.00	rc202	34.15	2.72	70.29	0.00
rc103	35.61	10.19	1.41	6.51	rc203	8.79	3.17	6.78	2.67
rc104	4.91	11.87	2.08	1.07	rc204	37.85	18.60	46.39	0.00
rc105	19.97	14.47	0.00	0.00	rc205	17.20	4.15	16.82	0.49
rc106	3.10	1.87	0.00	0.00	rc206	18.29	4.87	1.74	0.51
rc107	7.86	2.54	2.16	0.39	rc207	55.86	4.95	5.62	6.62
rc108	9.35	5.20	5.54	0.00	rc208	39.08	16.43	0.00	25.01

The results given by [27] is specific to the OP and the non-dominated sets they provided online are the best ones over 20 runs. Therefore, to compare our results with theirs, the number of individuals in our algorithm is set to one and the PF approximation obtained by CPMOEA/D is the non-dominated vectors generated over 5 runs. Since there are no reported results where the number of individuals is equal to or greater than 2, the obtained results for TOPTW with 2, 3, and 4 individuals are not presented here, and they are available upon request. Figs. 1 and 2 show 8 representative instances to compare the approximations of the PF obtained by



(a) c102 (b) c205

(c) r104 (d) rc108

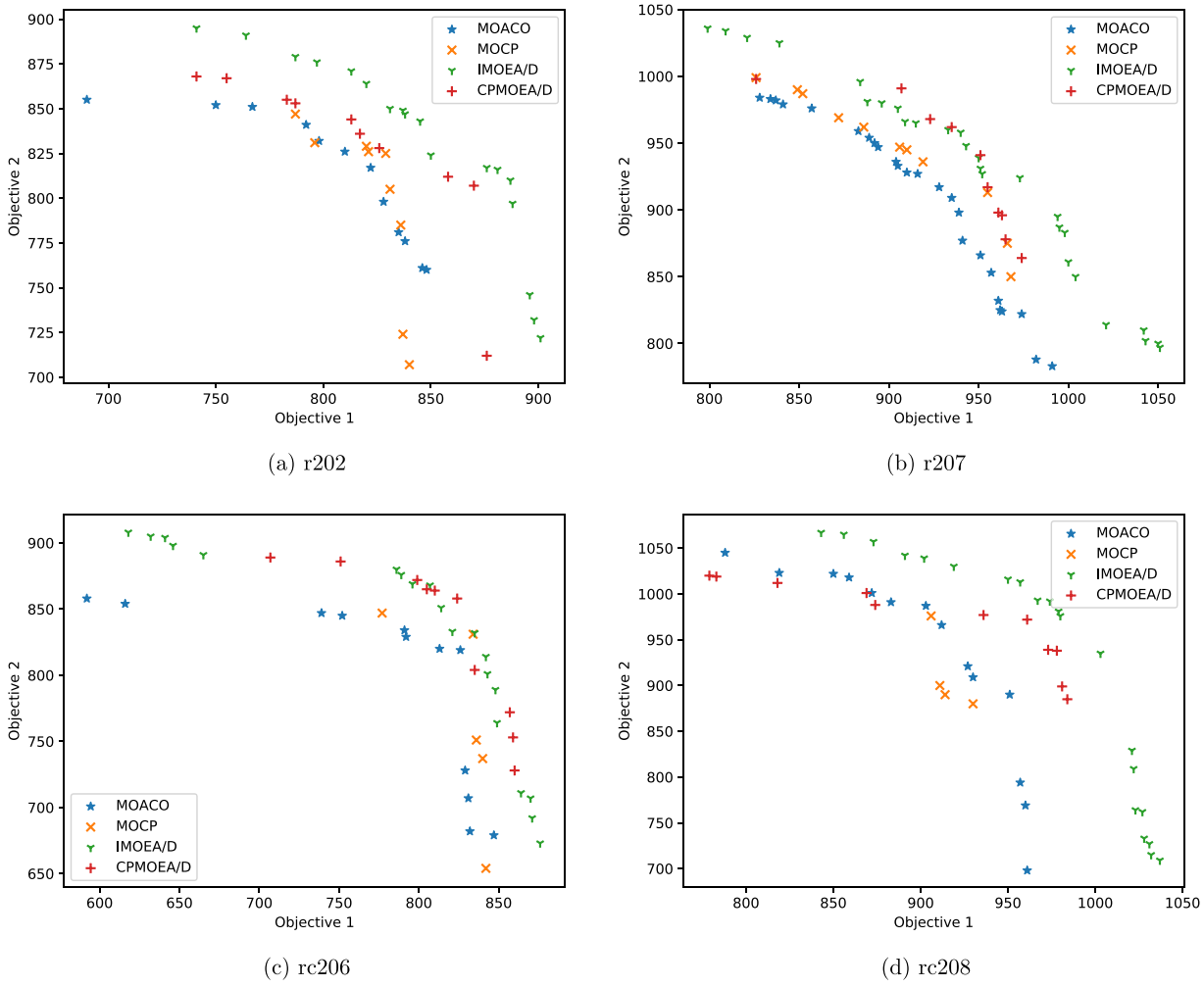
Fig. 1. Approximations of the PF generated by MOACO, MOCP, IMOEA/D, and CPMOEA/D.

four different algorithms. Tables 1–4 present the values of four performance metrics of all instances, respectively.

As shown in Table 1, MOCP and IMOEA/D have better performance of SC than MOACO in 34 instances (61%) and 36 instances (64%), respectively. Compared with MOACO in Table 2, more non-dominated objective vectors are obtained by MOCP and IMOEA/D in 29 instances (52%) and 38 instances (68%), respectively. Furthermore, when the MOEA/D framework is integrated with CP, the proposed CPMOEA/D has the best performance in 45 instances (80%) and 41 instances (73%) in terms of SC and NS, respectively. MOACO is the best method in only 1 instance (2%) on SC and 0 instance on NS. The PF approximations of four representative instances are shown in Fig. 1. Obviously, based on the TCH decomposition approach, the multi-objective optimization problem MOTOPTW is decomposed into a set of problems with single-objective, and it is possible to take full advantage of CP's powerful ability for solving routing-related problems. However, it is necessary to modify the subproblems' objectives of a general MOEA/D with TCH decomposition, and the results demonstrate the effectiveness of our modification. Finally, it is interesting to see that IMOEA/D performs the best in all the instances where CPMOEA/D has bad performance. Four such instances illustrate this point in Fig. 2. These instances are the ones with remote points and longer time windows, which results in less constraints and more feasible solution candidates. It is not beneficial to domain reduction and constraint propagation, two main methods implemented in CP.

The performance metrics in Tables 3 and 4 are used to evaluate the distribution of the approximations to the PF. CPMOEA/D outperforms others only in 19 instances (34%) on SP but obtains the best results in 41 instances (73%) on IGD. As discussed above, the IMOEA/D generally performs well on instances where the results of CPMOEA/D are not the best. In 51 instances (91%) CPMOEA/D or IMOEA/D obtained the best IGD. MOACO has the best results of SP in 16 instances (29%) but only 1 instance with the best IGD. It is not difficult to find that most of these objective vectors produced by MOACO are far away from the PF although they are well-distributed. Moreover, it is clear that CPMOEA/D shows better results than MOCP on these two metrics. The reason is that although searching with a starting point is an efficient strategy for the CP optimizer, it is observed that using current best solutions as starting points tends to result in no any improvement. By contrast, good but not the best solutions may give the CP optimizer more chances to search in different sections of the decision space. Therefore, it is more likely to obtain better solutions. The coding and decoding approaches presented above can provide an appropriate initial solution no matter how many individuals are involved. The results indicate that based on the decomposition method and good starting solutions CP optimizer can find well distributed non-dominated objective vectors. Also, although less time is consumed (about 350 s), CPMOEA/D finds better results on the first objective in 6 instances compared with the results given in [15] within 1800 s.

Finally, we can say that MOEA/D based multi-objective evolutionary algorithms presented above are competitive approaches



(a) r202 (b) r207

(c) rc206 (d) rc208

Fig. 2. Approximations of the PF generated by MOACO, MOCP, IMOEA/D, and CPMOEA/D.

for the MOTOPTW. The CPMOEA/D is a better choice for the MOTOPTW, especially for instances with clustered points and short time windows. For instances with remote points and longer time windows, the IMOEA/D is also recommended.

5. Conclusion

To the best of our knowledge, few studies are reported to solve the MOTOPTW where each checkpoint is associated with multiple profits. In this paper, we developed an MOEA/D and CP based multi-objective optimization algorithm for the MOTOPTW. This approach can take full advantage of the ability of MOEA/D to tackle multi-objective problems based on decomposition and the power of CP to deal with combinatorial optimization problems with multiple constraints. Coding and decoding approaches specific to the MOTOPTW are developed, and CP models with objectives specific to subproblems of MOEA/D are presented. The numerical results conducted on benchmark instances show that the CPMOEA/D is a promising approach to solve the MOTOPTW. Compared with the best-known approximations of PFs, a large number of new non-dominated objective vectors are found.

A significant future investigation area is the MOTOPTW with stochastic profits and resource consideration. It means that if an auditor visits a city and collects the profits, he or she may want to change the tour because in the middle of the tour he/she realizes that the gained profit is too low in comparison to the target profit level, in a probabilistic sense, it may be more suitable to take a risk and to visit checkpoints with high-profit variance. Another possibility for extension is to solve the MOTOPTW in the multiple-period in which new demand checkpoints arrive periodically. Moreover, in the multiple-period case, the number of new checkpoints in each period may be deterministic or stochastic (See [44] for OP with stochastic profits).

Acknowledgments

This research is partially supported by the Key and General Program of the National Natural Science Foundation of China (No. 51734004 and No. 51474044) and China Scholarship Council. Wanzhe Hu would like to thank Prof. Zhong Zheng for her support and help. P.M. Pardalos is supported by the Paul and Heidi Brown Preeminent Professorship at Industrial and Systems Engineering, University of Florida. Mahdi Fathi would like to thank Prof. Murray Brown, Mrs. Helen Brown at GH, Prof. Ahuja and Dr. Jha at Optym for their encouragement, mentoring, and support during this research.

References

[1] T. Tsiligirides, Heuristic methods applied to orienteering, J. Oper. Res. Soc. 35 (9) (1984) 797–809.
[2] S.E. Butt, T.M. Cavalier, A heuristic for the multiple tour maximum collection problem, Comput. Oper. Res. 21 (1) (1994) 101–111.
[3] M. Gendreau, G. Laporte, F. Semet, A branch-and-cut algorithm for the undirected selective traveling salesman problem, Networks 32 (4) (1998) 263–273.

[4] E.M. Arkin, J.S. Mitchell, G. Narasimhan, Resource-constrained geometric network optimization, in: Proceedings of the Fourteenth Annual Symposium on Computational Geometry, ACM, 1998, pp. 307–316.
[5] D. Feillet, P. Dejax, M. Gendreau, Traveling salesman problems with profits, Transp. Sci. 39 (2) (2005) 188–205.
[6] P. Vansteenwegen, W. Souffriau, D. Van Oudheusden, The orienteering problem: a survey, European J. Oper. Res. 209 (1) (2011) 1–10.
[7] A. Gunawan, H.C. Lau, P. Vansteenwegen, Orienteering problem: a survey of recent variants, solution approaches and applications, European J. Oper. Res. 255 (2) (2016) 315–332.
[8] P. Vansteenwegen, D. Van Oudheusden, The mobile tourist guide: an OR opportunity, OR insight 20 (3) (2007) 21–27.
[9] H. Tang, E. Miller-Hooks, A tabu search heuristic for the team orienteering problem, Comput. Oper. Res. 32 (6) (2005) 1379–1407.
[10] W. Souffriau, P. Vansteenwegen, G. Vanden Berghe, D. Van Oudheusden, The multiconstraint team orienteering problem with multiple time windows, Transp. Sci. 47 (1) (2013) 53–63.
[11] S.-W. Lin, F.Y. Vincent, A simulated annealing heuristic for the multiconstraint team orienteering problem with multiple time windows, Appl. Soft Comput. 37 (2015) 632–642.
[12] F.Y. Vincent, P. Jewpanya, C.-J. Ting, A.P. Redi, Two-level particle swarm optimization for the multi-modal team orienteering problem with time windows, Appl. Soft Comput. 61 (2017) 1022–1040.
[13] S.-W. Lin, F.Y. Vincent, Solving the team orienteering problem with time windows and mandatory visits by multi-start simulated annealing, Comput. Ind. Eng. 114 (2017) 195–205.
[14] H. Tae, B.-I. Kim, A branch-and-price approach for the team orienteering problem with time windows, Int. J. Ind. Eng. 22 (2) (2015) 243–251.
[15] R. Gedik, E. Kirac, A.B. Milburn, C. Rainwater, A constraint programming approach for the team orienteering problem with time windows, Comput. Ind. Eng. 107 (2017) 178–195.
[16] R. Montemanni, L.M. Gambardella, An ant colony system for team orienteering problems with time windows, Found. Comput. Decision Sci. 34 (4) (2009) 287.
[17] L.M. Gambardella, R. Montemanni, D. Weyland, Coupling ant colony systems with strong local searches, European J. Oper. Res. 220 (3) (2012) 831–843.
[18] P. Vansteenwegen, W. Souffriau, G.V. Berghe, D. Van Oudheusden, Iterated local search for the team orienteering problem with time windows, Comput. Oper. Res. 36 (12) (2009) 3281–3290.
[19] A. Gunawan, H.C. Lau, P. Vansteenwegen, K. Lu, Well-tuned algorithms for the team orienteering problem with time windows, J. Oper. Res. Soc. 68 (8) (2017) 861–876.
[20] F. Tricoire, M. Romauch, K.F. Doerner, R.F. Hartl, Heuristics for the multi-period orienteering problem with multiple time windows, Comput. Oper. Res. 37 (2) (2010) 351–367.
[21] S.-W. Lin, F.Y. Vincent, A simulated annealing heuristic for the team orienteering problem with time windows, European J. Oper. Res. 217 (1) (2012) 94–107.
[22] N. Labadie, R. Mansini, J. Melechovskỳ, R.W. Calvo, The team orienteering problem with time windows: an lp-based granular variable neighborhood search, European J. Oper. Res. 220 (1) (2012) 15–27.
[23] Q. Hu, A. Lim, An iterative three-component heuristic for the team orienteering problem with time windows, European J. Oper. Res. 232 (2) (2014) 276–286.
[24] T. Cura, An artificial bee colony algorithm approach for the team orienteering problem with time windows, Comput. Ind. Eng. 74 (2014) 270–290.
[25] M. Schilde, K.F. Doerner, R.F. Hartl, G. Kiechle, Metaheuristics for the bi-objective orienteering problem, Swarm Intell. 3 (3) (2009) 179–201.
[26] R. Martín-Moreno, M.A. Vega-Rodríguez, Multi-objective artificial bee colony algorithm applied to the bi-objective orienteering problem, Knowl.-Based Syst. 154 (2018) 93–101.
[27] Y.-H. Chen, W.-J. Sun, T.-C. Chiang, Multiobjective orienteering problem with time windows: an ant colony optimization algorithm, in: Technologies and Applications of Artificial Intelligence (TAAI), 2015 Conference on, IEEE, 2015, pp. 128–135.
[28] J. Wang, T. Weng, Q. Zhang, A two-stage multiobjective evolutionary algorithm for multiobjective multidepot vehicle routing problem with time windows, IEEE Trans. Cybern., preprint on webpage at https://ieeexplore.ieee.org/abstract/document/8338097/.
[29] T.-C. Chiang, W.-H. Hsu, A knowledge-based evolutionary algorithm for the multiobjective vehicle routing problem with time windows, Comput. Oper. Res. 45 (2014) 25–37.
[30] P.M. Pardalos, M.G. Resende, Handbook of Applied Optimization, Oxford University Press, 2002, p. xx,1095.
[31] A. Chinchuluun, P.M. Pardalos, A survey of recent developments in multiobjective optimization, Ann. Oper. Res. 154 (1) (2007) 29–50.
[32] Q. Zhang, H. Li, MOEA/D: A multiobjective evolutionary algorithm based on decomposition, IEEE Trans. Evol. Comput. 11 (6) (2007) 712–731.
[33] Y. Qi, Z. Hou, H. Li, J. Huang, X. Li, A decomposition based memetic algorithm for multi-objective vehicle routing problem with time windows, Comput. Oper. Res. 62 (2015) 61–77.
[34] L. Ke, Q. Zhang, R. Battiti, MOEA/D-ACO: A multiobjective evolutionary algorithm using decomposition and antcolony, IEEE Trans. Cybern. 43 (6) (2013) 1845–1859.
[35] M. Ehrgott, X. Gandibleux, A survey and annotated bibliography of multiobjective combinatorial optimization, OR-Spektrum 22 (4) (2000) 425–460.
[36] B. Gökgür, B. Hnich, S. Özpeynirci, Parallel machine scheduling with tool loading: a constraint programming approach, Int. J. Prod. Res. (2018) 1–17.
[37] H. Bouly, D.-C. Dang, A. Moukrim, A memetic algorithm for the team orienteering problem, 4OR 8 (1) (2010) 49–70.
[38] R. Sadykov, F. Vanderbeck, Bin packing with conflicts: a generic branch-and-price algorithm, INFORMS J. Comput. 25 (2) (2013) 244–255.
[39] C. Bazgan, H. Hugot, D. Vanderpooten, Solving efficiently the 0–1 multi-objective knapsack problem, Comput. Oper. Res. 36 (1) (2009) 260–279.
[40] G. Righini, M. Salani, Decremental state space relaxation strategies and initialization heuristics for solving the orienteering problem with time windows with dynamic programming, Comput. Oper. Res. 36 (4) (2009) 1191–1203.
[41] M.M. Solomon, Algorithms for the vehicle routing and scheduling problems with time window constraints, Oper. Res. 35 (2) (1987) 254–265.
[42] G.G. Yen, Z. He, Performance metric ensemble for multiobjective evolutionary algorithms, IEEE Trans. Evol. Comput. 18 (1) (2014) 131–144.
[43] P. Larranaga, C.M.H. Kuijpers, R.H. Murga, I. Inza, S. Dizdarevic, Genetic algorithms for the travelling salesman problem: a review of representations and operators, Artif. Intell. Rev. 13 (2) (1999) 129–170.
[44] T. Ilhan, S.M. Iravani, M.S. Daskin, The orienteering problem with stochastic profits, IIE Trans. 40 (4) (2008) 406–421.

International Journal of Industrial Engineering, 25(6), 828-842, 2018

MODELLING AND DECISION SUPPORT SYSTEM FOR INTELLIGENT MANUFACTURING: AN EMPIRICAL STUDY FOR FEEDFORWARD-FEEDBACK LEARNING-BASED RUN-TO-RUN CONTROLLER FOR SEMICONDUCTOR DRY-ETCHING PROCESS

Marzieh Khakifirooz[1], **Mahdi Fathi**[2], **and Chen-Fu Chien**[1, *]

[1]Department of Industrial Engineering and Engineering Management, National Tsing Hua University
Hsinchu, Taiwan
[*]Corresponding author's e-mail: cfchien@mx.nthu.edu.tw

[2]Department of Industrial and Systems Engineering, University of Florida
Gainesville, FL, United States

Shrinkage in semiconductor devices affects the process window of all wafer fabrication steps including plasma etching. Drifts or shifts are most significant effects on the etching process due to shrinkage in semiconductor devices. Any drift or shift affects on critical dimensions (CD) of the wafer and changes the thickness and the width over time. Therefore, there would be an essential need for estimation and minimization of CD variation on a wafer-to-wafer basis by optimization techniques. This study aims to design a learning-based control system for monitoring the CD in Dry-Etching process. Feedforward-feedback control technique is used to reduce CD variation. Among all learning-based control systems, the Iterative Learning Control (ILC) integrated with Virtual Meteorology (VM) data, as a well-known system which can involve both feedforward signal from the past events, and feedback signals from the output of the current event is used to learn the behavior of the system and enhance the performance of the controller run-by-run. The proposed control model is optimized by gradient learning approach. The result is validated through the simulated study manipulated from empirical data and shows the advantage of the proposed feedforward-feedback learning controller than the common run-to-run exponentially weighted moving average (EWMA) control design.

Keywords: critical dimension; disturbance rejection controller; feedforward-feedback control system; iterative learning controller; decision support system; virtual metrology

(Received on September 15; Accepted November 05, 2018)

1. INTRODUCTION

Dry etching (DE) is one of the critical wafer fabrication processes for semiconductor devices to remove selected layers of photoresist materials and maintain the quality of critical dimension (CD). DE technology as a core part of semiconductor device fabrication, cannot avoid many difficulties to cope with the challenges associated with new circumstances regarding the future development directions of semiconductor manufacturing. In general, DE suffers from the following challenges:

- Rapid development of lithography technology from double patterning technology to quadruple patterning technology, which is required tighter control law for global and local distribution control of CD (Turkot *et al.*, 2017).
- Tremendous increase of difficulties for High Aspect Ratio Contacts (HARC) etching such as DRAM capacitor, VNAND channel hole etching, and Metal Contact (MC) etching (Tandou *et al.*, 2016).
- More demand on lot-to-lot, wafer-to-wafer, and tool-to-tool or even die-to-die process variation management in recent years (Chien *et al.*, 2016).
- Request for innovative approaches to achieve goals of the etching process while utilizing new material/new structure (Ghodssi and Lin, 2011).

Specifications for DE are typically a tradeoff among rate, directionality, uniformity, selectivity, pattern dependencies, damage, and cleaning of etching process (Sawin, 1994). These requirements are contradictory, and typically resulting in the loss of CD control process because of photoresist erosion or lifetime of etching tools. Therefore, designing an excellent CD control system is one of the essentials to achieve high etching rates.

This study aims to propose an optimal control system for monitoring CD after the etching process. The CD is measured by metrology tool and is compensated by modifying the setup parameters of the controller or replacing the etching tools before their life cycle. A feedback message is sent to the next run for pre-adjustment, and a feedforward message is transmitted from the previous chemical-mechanical polishing (CMP) for post-adjustment. Then, the input recipe is updated for the next run based on recently measured process data through the metrology tools.

In this paper, we drive a learning control design technique based on the frequency of measurement data that integrates feedforward and feedback control on the etch process. To the best of our knowledge, there is only limited studies investigated on the Advanced Process Control (APC) system for CD of DE process. Some significant contributions in controlling CD is summarized in Table 1. Most of the developed model from literature are based on the run-to-run (R2R) controller which has a fixed profile structure and is not suitable for CD of DE process. We will broadly discuss this phenomenon in Section 2.

Based on both feedback and feedforward control signal and the mixture characteristics of control variables (i.e., lifetime, width, depth) plus the high-mixed production plan and short lifetime of etching tools, we need an advanced learning control system to be adapted with dynamic structure of DE process. Hereupon these needs, we design a novel feedforward-feedback learning R2R control system, to support all obligations in controlling CD during DE process.

The proposed approach is developed in a way that can be utilize the feedback information from metrology step combined with feedforward information of the previous CMP step. In addition, to overcome the challenges of metrology delay and shortage of data, the proposed control system is designed based on the learning-based control process and frequent measurement system (iteration). The applications of learning-based controller can cover a wide range of operations in the semiconductor industry such as chemical vapor deposition (CVD) process (Xu *et al.*, 1999; Chen *et al.*, 2011), batch processing (Kim *et al.*, 2013a; Kim *et al.*, 2013b, and temperature uniformity control (Won *et al.*, 2017). The learning-based control design in this study is inspired by Iterative Learning Control (ILC) system for R2R control design where iteration data is generated by Virtual Metrology (VM) tool.

The remainder of this paper is organized as follows: Section 2 introduces the core challenges, construction of problems, and intellectual foundations of investigating DE fabrication processes in this study. Section 3 presents the core structure of learning-based controller and the augmented feedforward-feedback R2R for controlling the CD. Section 4 demonstrates the case study of manufacturing data. Finally, the paper will be concluded in Section 5.

Table 1. Related studies on advanced process control of CD of DE process

Reference	Control System	Control Objective	Main Contribution
El Chemali *et al.* (2003)	feedforward and feedback R2R	Kalman filtering	minimizing and modeling etch-rate disturbance using a model of relationship between toll-etching life time and CD
El Chemali *et al.* (2004)	feedback R2R	Kalman filtering	manipulating the inputs and estimate disturbances; control sidewall angle
Mao *et al.* (2007)	multiple-dimensional closed-loop feedback R2R	EWMA; Kalman filtering	analyzing the effect of model mismatch and the controller's sensitivity to unknown noise
Wu *et al.* (2008)	R2R	Nonlinear Multiple Exponential-Weight Moving-Average (NMEWMA); Dynamic Model-Tuning Minimum-Variance (DMTMV)	modeling the relationship between exposure dose and focus and CD
Yang *et al.* (2010)	R2R	-	monitoring photoresist parameters such as photoresist thickness, photoactive compound and CD in-situ and in real-time
Ngo *et al.* (2013)	Linear Model Predictive (LMP)	EWMA; Kalman filtering	manipulating etch time to estimate state variables
Chien *et al.* (2015)	feedforward R2R	Analysis of Variance (ANOVA)	determining tool affinity
Hsu and Wu (2016)	R2R	error-smoothing EWMA	compensating the process variation

2. PROBLEM DEFINITION AND IDENTIFICATION

2.1. Terminologies and Notations

The notation and terminologies used in this study are listed as follows:

j	The iteration index.
t	The process run index, $t \geq 1$.
T	The total number of instance at each iteration.
$\boldsymbol{u}_j(t)$	Vector of input variables for iteration j at run t.
$\boldsymbol{y}_j(t)$	Vector of process outputs for iteration j at run t.
$\boldsymbol{y}_d(t)$	Vector of desired process outputs at run t.
$d(t)$	Process disturbance at run t.
$\boldsymbol{e}_j(t)$	Vector of deviation from the desired output for iteration j at run t.
J_t	Cost function at run t.
FF	Difference between the output and target of pre-layer.
RF	Etching tool lifetime.
C_{FF}	Coefficient of linear regression between intercept and output of pre-layer.
C_{RF}	Coefficient of linear regression between intercept and tool lifetime.
$P(q)$	Rational function of learning-based controller.
$Q(q)$	Q-filter of learning-based controller.
$L(q)$	Learning function of learning-based controller.
q	Forward time-shift operator.

2.2. Semiconductor DE Process

The etching process removes material from areas identified by the lithography process, to create structures for functional use (see Figure 1). Etching is a critically important process which every wafer needs to undergo this step many times before its completion. There are two main types of etching, wet-etching (liquid-based etchants) and DE (plasma-based etchants).

In wet-etching, the wafers are immersed in a tank of chemical solution or etchant. In wet etching, the photoresist material is removed through chemical reaction between wafer surface and etchants. DE is the removal process of material in the absence of the solvent. In DE process, the etching materials such as gases or plasma remove the substrate layer by the physical reaction. The wet-etching has some limitations in its applicability including the large size of patterning, isotropic process (Ivanov, 2017), hazardous of chemical materials, long completion time, and combination with a subsequent rinse (Jörg *et al.*, 2018). Therefore, DE technique is more applicable in the wafer fabrication process than wet-etching.

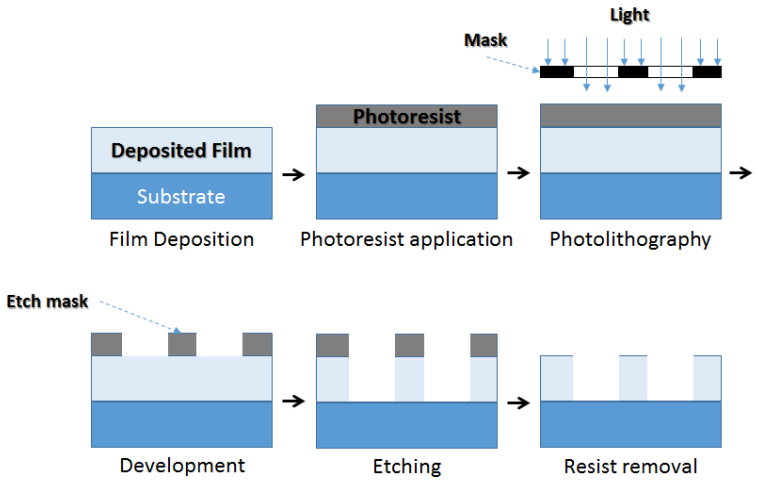


Figure 1. The etching process for wafer fabrication

Any etch process is characterized by certain properties and quality measurements as follows:

- Etching rate: The amount of material removed from the wafer over a defined period of time.
- Selectivity: The ability of the etch process to distinguish between the photoresist layer and substrate to be carved and the material not to be carved.
- Feature profile: Isotropic, etching proceeds at equal rates in both horizontal and vertical directions; Anisotropic, etching flows faster in one plane than in another.
- CD: dimensions of the delicate patterns formed on a wafer, i.e., width and depth.
- Residue: Remaining polymer after a post-etching cleaning process.
- Thickness: The photoresist thickness after the post-etching process.

In DE process, usually due to the lifetime of etching tools and mixed-product production process, it is challenging to obtain sufficient historical data as the reference information for controlling the production process. Therefore, the control system of DE process should design in a way that can be learned during short life-cycle of etching tools. In this study, among all properties, CD as the key characteristics of the etching process is selected for further investigations in designing the learning-based control system for DE.

2.3. Control System of CD

The CD is one of the important quality characteristics for wafer fabrication that its toleration should be continuously controlled, and keep it tight for yield enhancement. In semiconductor fabrication device, there is two different source of CD, the CD of scanned pattern via photolithography processes or photo-CD (PCD), and the CD measured by metrology tools after the etching process or etch-CD (ECD). The ECD represents the final line-width of fabricated patterns of each layer on the wafer. Thus, the ECD is the target for process control. The fundamental objective of the control process for ECD is minimizing the ECD's variation. Nevertheless, the ECD's variation is affected by the variability of both etching and photolithography processes. Therefore, the process control in etching should reduce the cumulative process variation from photolithography to the end of etching time. Controlling ECD is an inter-process R2R control (Chien *et al.*, 2015; Qin *et al.*, 2006) (see Figure 2), which requires the information from the past production step as well as measurement information from metrology tools to build a sufficient control system.

In the rest of this study for simplicity, the general CD refers to the ECD.

Showing in Figure 2, the inter-process R2R control deals with the process control of two or more inter-related process modules. In practice, although CVD and CMP are affecting the thin film thickness, the CMP processes can affect the control strategy of the CD. In particular, after the CMP process removes the unwanted photoresist materials, process engineers can report the photoresist thickness. The CMP measurement will be applied as a feedforward controller to the DE process. CD will be measured during the metrology step and will send feedback to the etching process (see Figure 3). As each wafer undergoes many times under DE for completing the fabrication process, therefore, the control purpose of CD is to minimize the effect of unmeasurable cumulative disturbance on the CD from the first layer of DE.

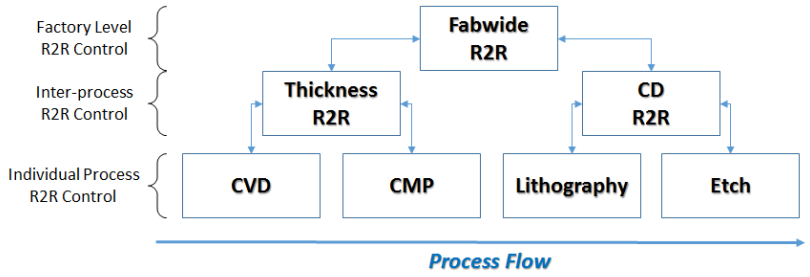


Figure 2. Classification the level of control process during the process flow of wafer fabrication (Chien *et al.*, 2015; Qin *et al.*, 2006)

In Figure 3, consider the system design of a controller which only contains the feedback signal from the metrology tools. Therefore, the linear dependency simply holds for input and output, and the relationship is usually known and formulated as:

$$\boldsymbol{y}(t) = slope * \boldsymbol{u}(t) + intercept \tag{1}$$

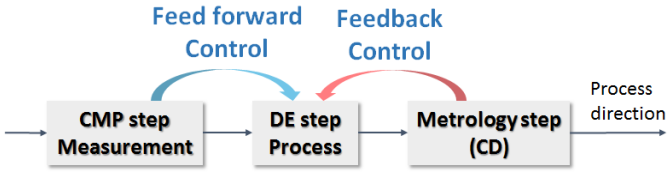


Figure 3. The process flow in system design of controller for CD

where the input $\boldsymbol{u}(t)$ is the etching time measuring the time of chemical or physical reaction to remove the photoresist material, the output $\boldsymbol{y}(t)$ is CD such as depth and width, and $intercept$ is cumulative disturbance and uncertainties. However, the process variation usually includes the disturbance $(d(t))$ from the effects of DE of previous photoresist layers (in short called pre-layer effect) or the etching tool lifetime in radio frequency (RF) hours. In this situation, both effects of pre-layer disturbance and tool lifetime can contribute into the "intercept". The effect of pre-layer disturbance could be reported from CMP measurement called the feedforward signal (see Figure 3). Therefore, the intercept in (1) is divided into two portions

$$intercept = C_{FF} * FF_{d(t)} + C_{RF} * RF_{time} + intercept' \tag{2}$$

and (1) is updated as follow:

$$\boldsymbol{y}(t) = slope * \boldsymbol{u}(t) + \left(C_{FF} * FF_{d(t)} + C_{RF} * RF_{time} + intercept'\right) \tag{3}$$

where C_{FF} is the coefficient of linear regression between intercept and output of pre-layer, $FF_{d(t)}$ is the difference between the output of pre-layer and target of pre-layer (or the disturbance from pre-layer), and C_{RF} is the coefficient of linear regression between intercept and tool lifetime. In practice, C_{FF} and C_{RF} are fixed, and if process engineers detect any change in CD, the coefficient parameters in the model will update to their optimal setting. This study aims to design a control system for controlling both feedback and feedforward signals to avoid any significant changes in the system and optimize the coefficient parameters without relying on the expert knowledge.

The sparse behavior of semiconductor manufacturing data (one observation per run), makes the R2R controller as the most applicable and suitable control design for this industry. However, for some process like DE, which is engaged with a high level of dynamicity the regular R2R controller is not efficient. On the other hand, in general design of R2R, first, input profiles for t-th run, named $\boldsymbol{u}(t)$, then the conventional R2R is used to update output $\boldsymbol{y}(t)$. Therefore, $\boldsymbol{y}(t)$ has no profile, while $\boldsymbol{u}(t)$ has a fixed profile structure. The fixed profile structure of R2R controller does not allow a different structure for input and output signal. In another word, the R2R controller is designed for the system when both input and output are describing the same characteristics. Therefore, regarding to the linear relationship between input and output of controller for

CD of DE as described in (3), when output is representing the CD and input is a function of etching time, etching tool's lifetime, and disturbance from pre-layer, obviously R2R controller is not the best choice for DE process.

In this study, we propose a learning-based control design for CD of DE to deal with the sparsity of semiconductor manufacturing data with a varying profile structure.

3. OPTIMIZATION BASED LEARNING IN CONTROL SYSTEM

The performance of a system which repeats a task multiple times can be improved through learning procedure from previous iterations. As the production repeats cyclically, at each loop/cycle/run the optimal decision is made and becomes the initial setting for the next loop/cycle/run. The control system can learn from this cyclic repetition and iteratively improve the performance accuracy. Learning control can deal with the problem of synthesizing an appropriate control input to make the system produce the desired action by repeated trails even with incomplete knowledge. In this study, we used the advantages of learning by repeating to support control system of the CD during the DE process.

There are many extensions of applying learning terminology in control theory which some of its advantages can be briefly classified as follows (Antsaklis, 2001):

1. To learn about the plant; how to derive new plant models and to learn how to incorporate changes.
2. To learn about the environment; this can be done using methods ranging from passive observation to active experimentation.
3. To learn about the controller; learn how to adjust specific control parameters to enhance performance.
4. To learn new design goals and constraints.

There is a variety of control strategy can be learned by historical information to design a new control system. In particular, the Iterative Learning Control (ILC) (Chen *et al.,* 2012), R2R control (Moyne *et al.,* 2000; Chien *et al.,* 2014), Adaptive Control (AC) (Åström and Wittenmark, 2013), Neural Networks (NN) (Hunt *et al.,* 1992), and Repetitive Control (RC) (Steinbuch, 2002) are commonly used methodology. Table 2, summarizes the applications and characteristics of the aforementioned learning-based control model. In this study, we design a hybrid learning-based feedforward-feedback control system to optimize C_{FF} in (3) based on the problem definition in Section 2. The proposed control system can compensate limitations of R2R controller with the following properties:

1. Automatic coefficient optimization of disturbance.
2. Robustness improvement through the use of causal feedback of metrology tools and feedforward of data from previous CMP step.
3. Iterative learning procedure to deal with lack of historical information for learning.

Table 2. Comparison key characteristics and objectives of ILC, R2R control, Adaptive Control, Neural Networks, and Repetitive Control system

Control System	Key Characteristics	Advantage	Limitation
ILC (Bristow *et al.*, 2006)	• time-based function • can be applied for dynamic batch processing • is built upon feedback and feedforward controller • can be built based on state space model • modifies the control input/signals • intended for discontinues operation • the initial setting is fixed for entire of process	• input has varying profile • output has frequent measurement • feedforward control can eliminate the lag in the transient tracking of feedback control • does not need the distribution of repeating disturbances • highly robust to system uncertainties	• has closed loop structure • friction, unmodeled nonlinear behavior, and disturbances can limit the effectiveness of feedforward control • cannot provide perfect tracking in every situation

R2R (Tan *et al.*, 2015)	• time-based function • can be applied for static model • can be built upon on state space model	• has close loop and open loop structure	• only a single product is manufactured on a single tool • input has fixed profile structure • output has sparse measurement
RC (Wang *et al.*, 2009)	• time-based function • can be applied for dynamic continuous process with periodic input • can be built upon transfer function • the initial setting is based on last trail	• input has varying profile • output has frequent measurement	• has single close loop structure • intended for continues operation
NN	• can be applied for nonlinear network • modifies the control parameters	• solve problems that do not have an algorithmic solution	• requires extensive training data • convergence rate is slow
AC	• modifies the control system	-	• does not use the historical data

3.1. Framework of an Intelligent Control System

To design a robust control system based on feedforward-feedback learning-based structure an intelligent control system is demanded. The framework of an intelligent control system is partitioned into three main parts: plant, data management center, and optimal controller where all three components continuously are connected to decision support system. The schematic of an intelligent control system is illustrated in Figure 4.

In the first part or production plant, the information is produced, and collected, then process engineers apply the decision rules. The entire of demanded information including metrology data, control parameters, scheduling and recipe information, and environmental factors are collected in this part and then stored in the data management center for further investigation.

Data management center is a vital part for statistical process control (i.e., fault detection, recipe management, and yield enhancement). In fact, the data management center is a feeding part of decision support system. The whole information, from raw data, decision rules or control law could be restored in the data management center.

The control system plays the analyzer role. Regardless the structure of the control system, all controllers are using the information from data management center, and again sending the control law and predicted information to there, where the decision support system can make the decision rules for plant performance enhancement.

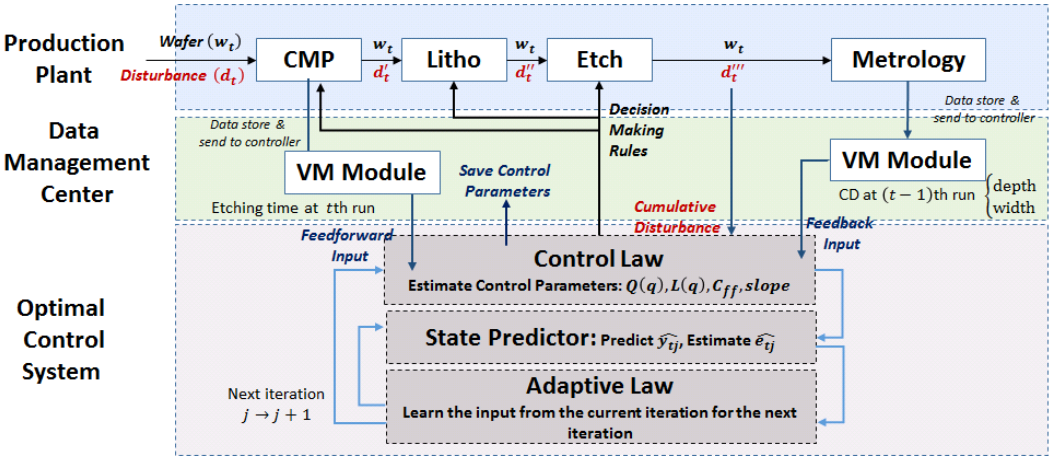


Figure 4. The framework of feedforward-feedback control system equipped with VM.

3.2. Feedforward-Feedback Learning-Based Control System

Consider the regular R2R controller which is equipped with only a feedback signal, as the most widely applied control system in semiconductor manufacturing. In controlling the CD of DE process, the regular feedback R2R controller cannot attain precisely near zero error due to time dependency of tool lifetime, tool heath, and cumulative disturbance that may transfer from pre-layers (Chien and Hsu, 2011; Chien *et al.*, 2014; Yu *et al.*, 2014).

To adapt the R2R controller with dynamic change in the system, there should be multiple measurements, however, the sparsely sampled output measurement from metrology tools can't support the dynamic change. One solution to overcome this weakness is to use the advantage of VM (Kang *et al.*, 2011; Tsai *et al.*, 2013; Baseman *et al.*, 2016; Jebri *et al.*, 2017) in the control system. The role of VM is to produce data for iterative control, therefore, the R2R controller can learn how to compensate the cumulative disturbance through iteration.

To adopt the VM into R2R controller we conduct the "just in time learning" approach (Cheng and Chiu, 2004) as the following steps:

1. At t-th run, virtual data is built upon the historical measured data through k-mean clustering.
2. Iteration is run using the data in the same cluster with the t-th measured data.
3. The process is separately conducted for both feedforward and feedback signals.

The proposed controller in Figure 4 is a controller that due to learning procedure can produce zero tracking error during repetitions of a command or eliminate the effects of a repeating disturbance on a control system output. Therefore, VM technology can be emerged into the R2R controller as a powerful tool to obtain models of imperfection and noisy data with a high degree of interpretability.

The next question to design a powerful control system for CD of DE process is how to eliminate the uncertainty using past performance information on the current trial (or how to bring the feedforward signal from CMP step into R2R control design)? The answer to this question can be given if the R2R controller will be formulated in a time-domain format.

Assume the discrete-time, linear time-invariant (LTI), Single-Input Single-Output (SISO) system as the principal structure for our proposed controller as follow:

$$\boldsymbol{y}_j(t) = P(q)\boldsymbol{u}_j(t) + d(t) \tag{4}$$

where q is the forward time-shift operator which means for any input signal at $(t+1)$-th run it can defined by the input signal at t-th run by $q.u(t) \equiv u(t+1)$ and the plant $P(q)$ is a proper rational function of q and has a delay, or equivalently relative degree of 1. We assume that $P(q)$ is asymptotically stable. Repeating disturbances (Boeren *et al.*, 2016), repeated nonzero initial conditions (Gal and bars, 2013), and systems augmented with feedforward-feedback control (Kuo, 2002) can be captured in $d(t)$. Therefore, with regards the desired system output $\boldsymbol{y}_d(t)$ at t-th run, the system performance or error signal can be defined as follows:

$$\boldsymbol{e}_j(t) = \boldsymbol{y}_d(t) - \boldsymbol{y}_j(t) \tag{5}$$

$$\boldsymbol{u}_j(t) = Q(q)\boldsymbol{u}_{j-1}(t) + L(q)\boldsymbol{e}_j(t) \tag{6}$$

where $Q(q) \in (0,1)$ is Q-filter (transforming the feedforward information) and $L(q)$ is the learning function (updating law). The dynamic control system with plant dynamics in (4) and learning dynamics in (6) are shown in Figure 5. By the definition of (4) and (6), the R2R controller is changed to the form of ILC.

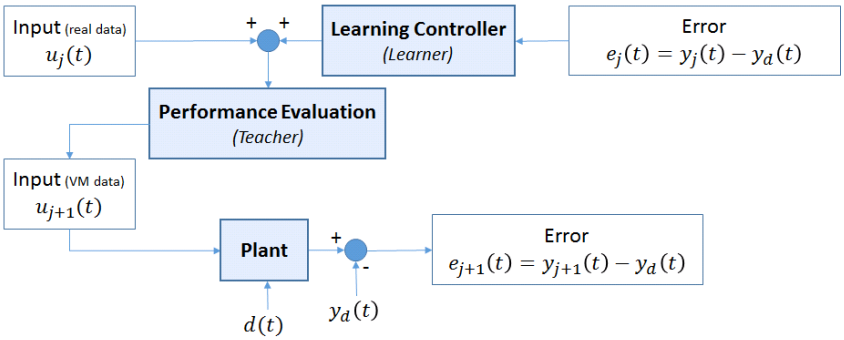


Figure 5. The block diagram of learning procedure of feedforward-feedback controller for one iteration.

3.3. Optimization the Control System for CD

The augmented feedforward-feedback learning-based controller similar to the regression model in (3) is formed into two parts:

1. A linear regression between CD and etching time.
2. The model of disturbances.

Therefore, the objective function of control system with regards to (3), (4), and (6) and the definition of error signal in (5) is:

$$\begin{aligned} &\min_{u(t)} J_t = \left|\boldsymbol{e}_j(t)\right|^2 \\ &\boldsymbol{e}_j(t) = \boldsymbol{y}_d(t) - \boldsymbol{y}_j(t) \\ &\boldsymbol{y}_j(t) = slope * \boldsymbol{u}_j(t) + d(t) \\ &\boldsymbol{u}_j(t) = Q(q)\boldsymbol{u}_{j-1}(t) + L(q)\boldsymbol{e}_j(t-1) \end{aligned} \tag{7}$$

where $d(t) = C_{FF} * FF_{d(t)} + C_{RF} * RF_{time}$.

Deriving the decision variables $slope$, C_{FF}, $Q(q)$ and $L(q)$ is necessary to solve the optimization problem in (7). There is infinite possible iterative procedures to solve the optimization problem in (7). The gradient approach (Owens *et al.*, 2009) has the most straightforward form and has been widely investigated in literature for optimizing the learning-based error (Amann *et al.*, 1996).

4. EMPIRICAL STUDY

Follow modeling a predictive control design for minimizing the variation of CD, the next step is to implement the proposed controller with process data and under unmeasurable disturbance. The primary objective of the controller is to regulate the CDs in the face of all sources of uncertainties, in which the difference between actual output and desired output will be minimized. Accordingly, if the control model can reject the effect of uncertainties, then the actual output and the input should be very close to each other.

In addition, for any control system design, it is essential to understand the system configuration, calibration, and initialization before system assembles in the real plant. In the case of learning-based feedforward-feedback control system, we should understand how the learning process can reject the disturbance and how the learning rate can transfer the disturbance-free output at the current run to the input at the next run. Furthermore, for the DE process, we would like to understand the effect of various process parameters on CD from the quality of photolithography or CMP process.

To estimate the validity of the proposed control system and the effect of learning in feedforward-feedback control design, we implement a simulation study for 200 lots of manufacturing data. To design the simulation scenario, we firstly collect empirical data, and the density plot of real data is illustrated in Figure 6. Manufacturing data in this study are accumulated the effect of etching tools' lifetime and etching time together. Therefore, equation (7) updates as follow:

$$\begin{aligned} &\min_{u(t)} J_t = \left|\boldsymbol{e}_j(t)\right|^2 \\ &\boldsymbol{e}_j(t) = \boldsymbol{y}_d(t) - \boldsymbol{y}_j(t) \\ &\boldsymbol{y}_j(t) = slope * \boldsymbol{u}_j(t) + C_{FF} * FF_{d(t)} \\ &\boldsymbol{u}_j(t) = Q(q)\boldsymbol{u}_{j-1}(t) + L(q)\boldsymbol{e}_j(t-1) \end{aligned} \tag{8}$$

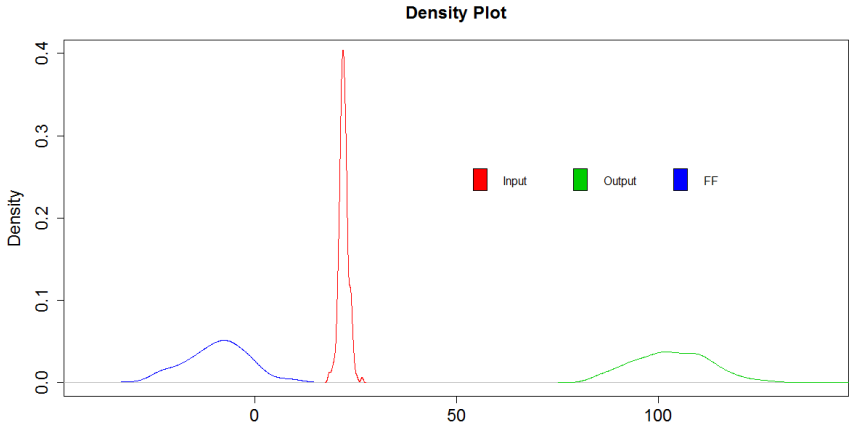


Figure 6. The density plot of empirical $\boldsymbol{u}(t)$ (red line), $\boldsymbol{y}(t)$ (green line), and $FF_{d(t)}$ (blue line)

The following steps design the simulation process for performance evaluation of the proposed feedforward-feedback learning-based R2R for controlling the CD in DE process for a SISO system.

Step 1: Consider $FF_{d(t)}$ in simulation design equivalent to the empirical data as illustrated in Figure 6. In addition, initiate $\boldsymbol{y}_1(1)$ and $\boldsymbol{u}_1(1)$ equal to the empirical result.

Step 2: Consider 200 data set $(\boldsymbol{y}(t), \boldsymbol{u}(t), FF_{d(t)})$ as the information for 8 lots.

Step 3: Set number of iteration $j = 20$.

Step 4: Initiate the parameter setting for $Q(q)$, $L(q)$, $slope$, and C_{FF} as (0.5,1,1,1), respectively.

Step 5: Set $\boldsymbol{y}_d(t)$ equal to the $\boldsymbol{u}_1(t-1)$.

Step 6: Generate virtual data by "just in time learning" approach as mentioned in Section 3.2, where parameters of k-mean clustering method are selected by tuning algorithm.

Step 7: Optimize the model in (8) by gradient decent method in Owens *et al.* (2009).

Step 8: Adopt Residual Mean Square Error (RMSE) in (9), and Range in (10) for performance comparison between the feedforward-feedback learning-based R2R controller and empirical data.

$$RMSE = \sqrt{\frac{\sum_{t=1}^{200} |\boldsymbol{e}(t)|^2}{t}} \tag{9}$$

$$Range = \max_t \boldsymbol{y}(t) - \min_t \boldsymbol{y}(t) \tag{10}$$

Figure 7 illustrates the effect of iteration on the performance of the learning control system for the first 11 iterations of the simulation scenario. It is clear that always a high number of repetition cannot guarantee the better performance. Among all situations, $j = 7$ and $j = 8$ perform the best result in case of Range and RMSE, respectively. In addition, error reaches to the steady-state condition after 8-th iterations. Table 3 summarizes the effect of RMSE and Range for each iteration after 200 runs.

Table 3. RMSE, and Range for each iteration

Index	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$	$j = 6$	$j = 7$	$j = 8$	$j = 9$	$j = 10$
$RMSE$	-	960	472	221	107	44	26	19	24	31
$Range$	-	1710	797	370	157	82	52	75	103	148
Index	$j = 11$	$j = 12$	$j = 13$	$j = 14$	$j = 15$	$j = 16$	$j = 17$	$j = 18$	$j = 19$	$j = 20$
$RMSE$	41	53	68	85	107	133	164	202	249	306
$Range$	198	255	331	425	543	682	853	1062	1318	1640

Figure 7 shows that how iteration can be helpful for controller to learn and eliminate the effect of unmeasurable disturbance. As we can see for $j = 2$, error is calculated by $\boldsymbol{e}_1(t) = \boldsymbol{u}_1(t) - \boldsymbol{y}_2(t)$ and due to the initial setting of parameters

defined in **Step 4**, the cumulative error is positive. Since the estimated error $\boldsymbol{e}(t)$ in comparison with the estimated input $\boldsymbol{u}(t)$ is very small (see Figure 8), therefore, the effect of input is more stronger than the effect of error on estimating the output $\boldsymbol{y}(t)$. In this example, as we can see on Figure 7 the value of $\boldsymbol{u}(t)$ is always positive, therefore, it is expected that error has positive value at the first iteration is expected. For the second iteration $\boldsymbol{y}_d(t) = \boldsymbol{u}_1(t)$ remains fix, however, $\boldsymbol{y}_j(t)$ is indirectly affected by $\boldsymbol{u}_{j-1}(t)$ and this causes that $\boldsymbol{e}_2(t)$ has opposite sign of $\boldsymbol{e}_1(t)$. This pattern is repeated until learning algorithm (iterations) can eliminate the effect of unmeasurable disturbance.

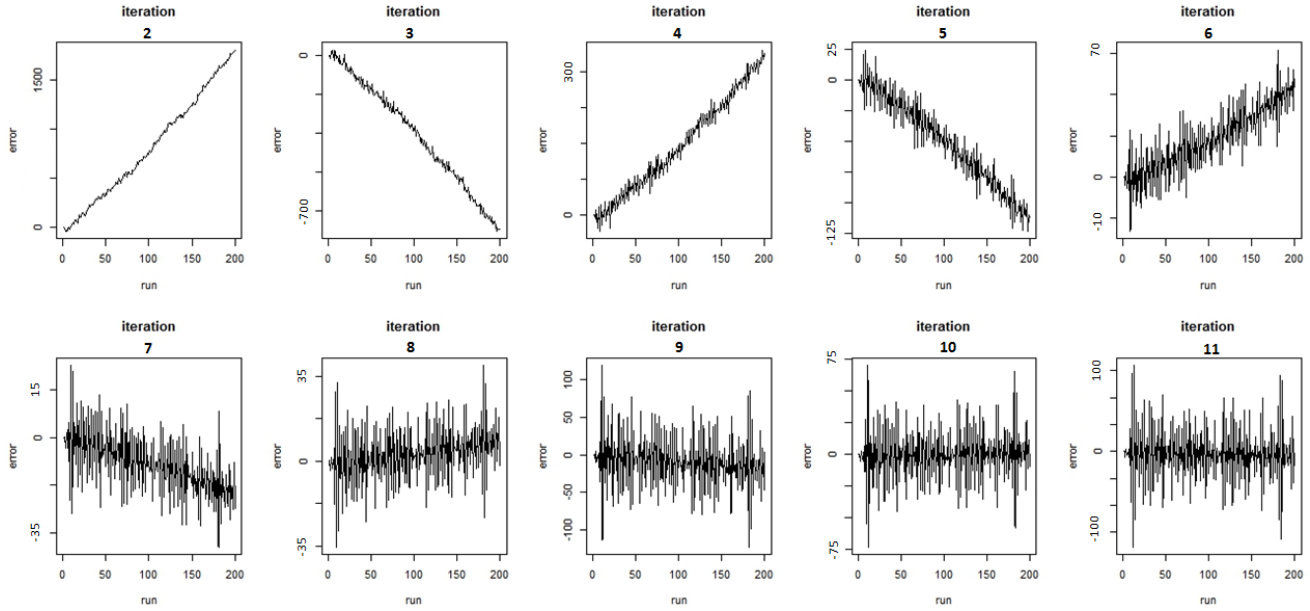


Figure 7. The effect of iteration on optimization of (8).

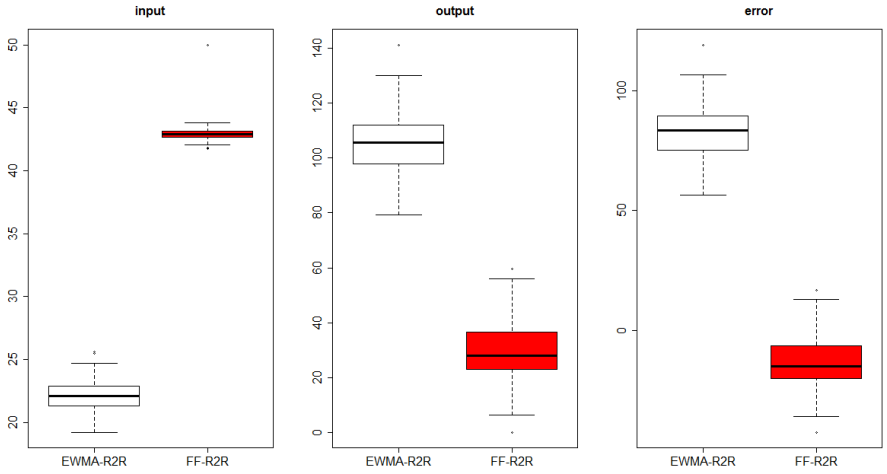


Figure 8. Performance comparison between proposed R2R controller and empirical data, where the y-axis indicates $\boldsymbol{u}(t)$, $\boldsymbol{y}(t)$, and $\boldsymbol{e}(t)$ for each plot from left to right, respectively.

Figure 8, shows the comparison results between empirical data (EWMR-R2R with fixed discount factor 0.3) and feedforward-feedback learning-based R2R. The results indicate that the proposed R2R controller tightens up the excellent performance bound, and eventually achieves a lower cost, together with an extensive disturbance, in comparison with the empirical control design (EWMR-R2R with fixed discount factor 0.3). In total, although in contrast, the proposed R2R controller has weakness in the performance improvement for Range (see Table 4), it can be taught to the system to estimate the output very close to the input which resulted in improvement of the error.

As discussed in Amann *et al.* (1996), the feedforward-feedback learning-based R2R algorithm employed in this study can be implemented in practice if and only if the feedback loop is available. For implementation, the free parameters Q, and L in (8) must be chosen appropriately. The parameter L is related to the size of the error, and the parameter Q to the size of the change of the input. Therefore the sensitivity analysis is essential for study the speed of convergence.

Table 4. RMSE, and Range for each iteration

Index	RMSE		Range	
	EWMA-R2R	Learning-R2R	EWMA-R2R	Learning-R2R
$\boldsymbol{e}(t)$	82.147	19.051	42.75	75.68
$\boldsymbol{u}(t)$	22.2	4.37	5.579	4.88
$\boldsymbol{y}(t)$	104.21	22.43	40.74	77.97

To illustrate the effect of Q and L, we could consider Q to be fixed and set $L = aQ$, $a > 0$. Therefore, for small a the algorithm is expected to change the incremental input substantially to achieve a small error and causing a fast rate of decrease of J_t. Contrariwise, for large a the convergence rate of J_t will hold with a slow rate in decreasing pattern.

5. CONCLUSION

Smart decision support system is critical for intelligent manufacturing, integrating operations research modeling, optimization, big data analytics, and AI to empower flexible decisions with complicated objectives involved in strategic, operational, and tactical decisions. For a process that is repetitive or cyclic, the learning type control method should be the first choice for control. The specific type of learning type control should then be selected according to the characteristics of the process. In this study regards to the cycling nature of the semiconductor manufacturing and the different profiles of input variables, the hybrid feedforward-feedback learning-based R2R control system is selected for process monitoring of DE process.

The feedforward-feedback learning-based R2R system has proven to be accurate and flexible for monitoring the CD during the DE process. This approach has modeled wafer fabrication processes with the low error for empirical data compare to EWMA-R2R control design. The presented methodology is able to learn from the input variable and ignore the effect of unmeasurable uncertainties through the iterative learning process and the help of virtual data, and compensate the variation of CD.

The constraint-free optimization algorithm in (8) is evolved by gradient learning approach and has the capability to work online and offline for the supervisory control plant. However, to facilitate the optimization algorithm, the system can be modeled firstly by historical data in the offline mode to initiate the parameter setting for online mode. Regards to the data-warehouse management strategy, the offline model can also store the feedforward/feedback signal in the data warehouse till receiving the feedback/feedforward signal then can optimize the system.

The result presented in this paper was regarding the capabilities of the feedforward-feedback learning-based R2R controller for LTI system. Furthermore, there are a number of extensions that could be considered as the future research direction. Future research can be done to employ big data analytics (e.g. Chien and Chuang, 2014; Khakifirooz *et al.*, 2018) to enhance CD control and the yield. Also, more studies can be done to address the issues of convergence rate and robustness of the proposed control system. As it is an often case in semiconductor manufacturing, there would be a potential of the technique for investigating the non-linear optimization model for CD instead of the linear model in (3). Also, similar control system can be considered with time-varying metrology delay of LTI system. There is a great deal of research needed for comparison in the area of learning-based control system, such as NN or kernel-based optimization control systems. The learning algorithm could be designed for learning the different source of disturbance (i.e., non-stationary disturbance or stationary disturbance) and the model in (3) could expand by the effect of the other environmental variables such as gas flows, and temperature. The optimization algorithm could involve the control variable with constraint and enhance the performance of the CD controller.

ACKNOWLEDGEMENTS

This research is supported by the Artificial Intelligence for Intelligent Manufacturing Systems (AIMS) Research Center of Ministry of Science and Technology, Taiwan (MOST 107-2634-F-007-002; MOST 107-2634-F-007-009).

REFERENCES

Amann N, Owens DH, Rogers E. Iterative learning control for discrete-time systems with exponential rate of convergence. IEE Proceedings-Control Theory and Applications. 1996 Mar;143(2):217-24.

Antsaklis PJ. Intelligent control. Wiley Encyclopedia of Electrical and Electronics Engineering. 2001 Aug 21.

Åström KJ, Wittenmark B. Adaptive control. Courier Corporation; 2013 Apr 26.

Baseman RJ, He J, Yashchin E, Zhu Y, inventors; International Business Machines Corp, assignee. Run-to-run control utilizing virtual metrology in semiconductor manufacturing. United States patent US 9,299,623. 2016 Mar 29.

Boeren F, Bareja A, Kok T, Oomen T. Frequency-domain ILC approach for repeating and varying tasks: With application to semiconductor bonding equipment. IEEE/ASME Transactions on Mechatronics. 2016 Dec 1;21(6):2716-27.

Bristow DA, Tharayil M, Alleyne AG. A survey of iterative learning control. IEEE Control Systems. 2006 Jun;26(3):96-114.

Chen J, Cheng N, Cheng YC. Hybrid model based iterative learning control for semiconductor processes with uncetain metrology delay. In Industrial Electronics and Applications (ICIEA), 2011 6th IEEE Conference on 2011 Jun 21 (pp. 1519-1524). IEEE.

Chen Y, Moore KL, Ahn HS. Iterative learning control. InEncyclopedia of the Sciences of Learning 2012 (pp. 1648-1652). Springer, Boston, MA.

Cheng C, Chiu MS. A new data-based methodology for nonlinear process modeling. Chemical Engineering Science. 2004 Jul 1;59(13):2801-10.

Chien CF, Chen YJ, Hsu CY. A novel approach to hedge and compensate the critical dimension variation of the developed-and-etched circuit patterns for yield enhancement in semiconductor manufacturing. Computers & Operations Research. 2015 Jan 1;53:309-18.

Chien CF, Chen YJ, Hsu CY, Wang HK. Overlay Error Compensation Using Advanced Process Control with Dynamically Adjusted Proportional-Integral R2R Controller. IEEE Transactions on Automation Science and Engineering. 2014;11(2), 473-484.

Chien CF, Chuang SC. A Framework for Root Cause Detection of Sub-batch Processing System for Semiconductor Manufacturing Big Data Analytics. IEEE Transactions on Semiconductor Manufacturing. 2014;27(4): 475-488.

Chien CF, Hsu CY. UNISON Analysis to Model and Reduce Step-and-Scan Overlay Errors for Semiconductor Manufacturing. Journal of Intelligent Manufacturing. 2011;22(3), 399-412.

El Chemali C, Freudenberg J, Hankinson M, Collison W, Ni T. Critical dimension control of a plasma etch process by integrating feedforward and feedback run-to-run control. Journal of Vacuum Science & Technology B: Microelectronics and Nanometer Structures Processing, Measurement, and Phenomena. 2003 Nov;21(6):2304-12.

El Chemali C, Freudenberg J, Hankinson M, Bendik JJ. Run-to-run critical dimension and sidewall angle lithography control using the PROLITH simulator. IEEE Transactions on Semiconductor Manufacturing. 2004 Aug;17(3):388-401.

Gal PL, Bars ML. Rotating thermal flows in natural and industrial processes. Geophysical & Astrophysical Fluid Dynamics. 2013 Sep; 107(6): 720-723

Ghodssi R, Lin P, editors. MEMS materials and processes handbook. Springer Science & Business Media; 2011 Mar 18.

Hsu CY, Wu JZ. Error-smoothing exponentially weighted moving average for improving critical dimension performance in photolithography process. International Journal of Industrial Engineering. 2016 Sep 1;23(5).

Hunt KJ, Sbarbaro D, Żbikowski R, Gawthrop PJ. Neural networks for control systems—a survey. Automatica. 1992 Nov 1;28(6):1083-112.

Ivanov A. Silicon Anodization as a Structuring Technique: Literature Review, Modeling and Experiments. Springer; 2017 Sep 10.

Jebri MA, El Adel EM, Graton G, Ouladsine M, Pinaton J. Virtual Metrology applied in Run-to-Run Control for a Chemical Mechanical Planarization process. InJournal of Physics: Conference Series 2017 Jan (Vol. 783, No. 1, p. 012042). IOP Publishing.

Jörg T, Hofer AM, Köstenbauer H, Winkler J, Mitterer C. Oxidation and wet etching behavior of sputtered Mo-Ti-Al films. Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films. 2018 Mar;36(2):021513.

Kang P, Kim D, Lee HJ, Doh S, Cho S. Virtual metrology for run-to-run control in semiconductor manufacturing. Expert Systems with Applications. 2011 Mar 1;38(3):2508-22.

Kim H, Park JH, Lee KS. Methods and properties of quadratic iterative learning control for semi-conductor processes under different perturbations. In Control, Automation and Systems (ICCAS), 2013 13th International Conference on 2013a Oct 20 (pp. 570-573). IEEE.

Kim HT, Lee KW, Yang HJ, Kim SC. A self-learning method for automatic alignment in wafer processing. International Journal of Precision Engineering and Manufacturing. 2013b Feb 1;14(2):215-21.

Khakifirooz M, Chien CF, Chen YJ. Bayesian inference for mining semiconductor manufacturing big data for yield enhancement and smart production to empower Industry 4.0. Applied Soft Computing. 2018;68: 990-999.

Kuo AD. The relative roles of feedforward and feedback in the control of rhythmic movements. Motor control. 2002 Apr;6(2):129-45.

Mao ZJ, Kang W. Benchmark study of run-to-run controllers for the lithographic control of the critical dimension. Journal of Micro/Nanolithography, MEMS, and MOEMS. 2007 Apr;6(2):023001.

Moyne J, Del Castillo E, Hurwitz AM. Run-to-run control in semiconductor manufacturing. CRC press; 2000 Nov 30.

Ngo YS, Ang KT, Tay A. Method for real-time critical dimensions signature monitoring and control: Sensor, actuator, and experimental results. Review of Scientific Instruments. 2013 Jan;84(1):015116.

Owens DH, Hatonen JJ, Daley S. Robust monotone gradient-based discrete-time iterative learning control. International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal. 2009 Apr;19(6):634-61.

Qin SJ, Cherry G, Good R, Wang J, Harrison CA. Semiconductor manufacturing process control and monitoring: A fab-wide framework. Journal of Process Control. 2006 Mar 1;16(3):179-91.

Sawin HH. Challenges in dry etching: uniformity, selectivity, pattern dependencies, damage, and cleaning. Microelectronic Engineering. 1994 Jan 1;23(1-4):15-21.

Steinbuch M. Repetitive control for systems with uncertain period-time. Automatica. 2002 Dec 1;38(12):2103-9.

Tan F, Pan T, Li Z, Chen S. Survey on run-to-run control algorithms in high-mix semiconductor manufacturing processes. IEEE Transactions on Industrial Informatics. 2015 Dec;11(6):1435-44.

Tandou T, Kubo S, Negishi N, Izawa M. Improving the etching performance of high-aspect-ratio contacts by wafer temperature control: Uniform temperature design and etching rate enhancement. Precision Engineering. 2016 Apr 1;44:87-92.

Tsai PF, Tsen A, Sung JN, inventors; Taiwan Semiconductor Manufacturing Co (TSMC) Ltd, assignee. System and method for implementing a virtual metrology advanced process control platform. United States patent US 8,437,870. 2013 May 7.

Turkot B, Carson S, Lio A. Continuing Moore's law with EUV lithography. In Electron Devices Meeting (IEDM), 2017 IEEE International 2017 Dec 2 (pp. 14-4). IEEE.

Wang Y, Gao F, Doyle III FJ. Survey on iterative learning control, repetitive control, and run-to-run control. Journal of Process Control. 2009 Dec 1;19(10):1589-600.

Won W, Park K, Kim J. Combined iterative learning and delta-operator adaptive linear quadratic Gaussian control of a commercial rapid thermal processing system. Chemical Engineering Science. 2017 Dec 31;174:146-56.

Wu CF, Hung CM, Chen JH, Lee AC. Advanced process control of the critical dimension in photolithography. International Journal of Precision Engineering and Manufacturing. 2008 Jan;9(1):12-8.

Xu JX, Chen Y, Lee TH, Yamamoto S. Terminal iterative learning control with an application to RTPCVD thickness control. Automatica. 1999 Sep 1;35(9):1535-42.

Yang G, Ngo YS, Putra AS, Ang KT, Tay A, Fang ZP. Monitoring and control of photoresist properties and CD during photoresist processing. In Metrology, Inspection, and Process Control for Microlithography XXIV 2010 Apr 2 (Vol. 7638, p. 763828). International Society for Optics and Photonics.

Yu HC, Lin KY, Chien CF. Hierarchical Indices to Detect Equipment Condition Changes with High Dimensional Data for Semiconductor Manufacturing. Journal of Intelligent Manufacturing, 2014;25(5), 933-943.

Proc IMechE Part O: J Risk and Reliability
XX(X):1–10

DOI: 10.1177/ToBeAssigned
www.sagepub.com/
SAGE

Redundancy Allocation Problem in a Bridge System with Dependent Subsystems

Kamyar Sabri-Laghaie[1], Milad Eshkevari[2], Mahdi Fathi[3], and Enrico Zio[4]

Abstract

The Redundancy Allocation Problem (RAP) is an important problem in system reliability design. Many researchers have investigated the RAP under different assumptions and for various system configurations. However, most of the studies have disregarded the dependence among components and subsystems. In real-world applications, the performance of components and subsystems can affect each others. For instance, the heat radiated by a subsystem can accelerate degradation of adjacent components or subsystems. In this paper, a procedure is proposed for solving the RAP of a bridge structure with dependent subsystems. Copula theory is utilized for modeling dependence among subsystems, and artificial neural network (ANN) and particle swarm optimization (PSO) are applied for finding the best redundancy allocation. A numerical example is included to elaborate the proposed procedure and show its applicability.

Keywords

Redundancy allocation problem, Bridge system, dependence, Copula theory, Artificial neural network, Particle swarm optimization

1. Introduction

Today, survival of companies in the competitive markets strongly depends on the capability of effectively assigning to the customer needs of high performance and quality. Reliability is related to the ability of a system to meet the quality requirements. It is one of the most important factors in designing and manufacturing of products. In order to maximize system reliability, three strategies can be adopted: 1) enhancement of component reliability, 2) redundancy and, 3) combination of the two mentioned alternatives [14,23]. These strategies usually increase the demand for resources (cost, volume, weight, etc.). Therefore, at the phase of designing a highly reliable system, an important problem is to get the balance between reliability and other resource constraints.

The problem of maximizing system reliability through redundancy is called "redundancy allocation problem (RAP)". RAP has been vastly studied for different system structures, objective functions and time to failure distributions [5]. It is known that RAP is an NP-hard problem [6]. RAP is usually formulated as a non-linear integer programming problem, which is in general difficult to solve due to the considerable amount of computational effort required to find the exact solution. Hence heuristic and meta-heuristic approaches have been widely used to deal with this problem (e.g. Tabu Search [13], Genetic Algorithm [20], Particle Swarm Optimization [2]).

In real-world applications, system components may share some environmental factors such as temperature, pressure, load, etc. with each other. In other words, factors originated from some components may affect the performance of other components. Moreover, environmental factors may be shared among the subsystems of a system. For example, thermal radiation from a component or subsystem can impact the overall performance of other components or subsystems. Furthermore, the number of components installed in a subsystem can also determine the extent of dependence among subsystems. However, most of the RAP researches typically ignore the dependence among components or subsystems [17]. With regard to the literature, Kotz et al. [12] studied reliability when two components are positively quadrant dependent. For this aim, they used a number of bivariate distributions to model the dependent components and investigate the effect of components correlation on the lifetime of parallel redundant systems. Costa Bueno [7] used the reverse rule of order 2 property between component lifetimes to study the RAP of k-out-of-n systems via a martingale approach. He defined the concept of "minimal standby redundancy" and used it for allocating a redundant spare in a k-out-of-n:F system with dependent components. Belzunce et al. [3] used joint stochastic orders to study optimal allocation of redundant components in series and parallel systems with two dependent components. Belzunce et al. [4] studied optimal allocation of redundant components in series, parallel and k-out-of-n:F systems with more than two components. For this purpose, they extended bivariate joint stochastic orders and used multivariate joint stochastic hazard rate and reversed hazard rate orders to

[1]Faculty of Industrial Engineering, Urmia University of Technology, Urmia, Iran
[2]Faculty of Industrial Engineering Urmia University of Technology, Urmia, Iran
[3]Center for Applied Optimization, Department of Industrial and Systems Engineering, University of Florida, USA
[4]Dipartimento di Energia, Politecnico di Milano, Italy

Corresponding author:
Mahdi Fathi
Email: mahdi.fathi@ufl.edu

select redundant components. You and Li [26] studied RAP in engineering systems with dependent component lifetimes. They considered active and standby policies and built the likelihood ratio order and the hazard rate order for lifetimes in allocating redundancies to k-out-of-n systems. Gupta and Kumar [9] studied the problem of stochastic comparison of component and system redundancies where components are dependent and identically distributed. For this aim, likelihood ratio ordering, reversed failure rate ordering, failure rate ordering and the usual stochastic ordering were considered for carrying out the study. Further, Jeddi and Doostparast [10] studied optimal redundancy allocation problems in engineering systems with dependent component lifetime where no specific assumptions on the dependence structure of lifetimes are considered.

Despite the vast literature on RAP, only few researches have considered dependence among components and subsystems. Therefore, a procedure for evaluating RAP when subsystems are not independent is proposed in the current paper. Then, a bridge system is considered and the proposed procedure is applied to it. In brief, the redundancy allocation problem of a bridge system with dependence among subsystems is considered in this paper. The aim is to propose a procedure for the optimal allocation of components to a bridge system where the subsystems can be mutually dependent on each other. It is supposed that the parameters and characteristics of the components specify the type and extent of dependence among subsystems. A methodology based on Copula theory and Artificial Neural Network (ANN) is applied to model the dependence among subsystems. A Particle Swarm Optimization (PSO) Algorithm is employed to solve the dependent redundancy allocation problem.

The main contributions of the paper are as follows:

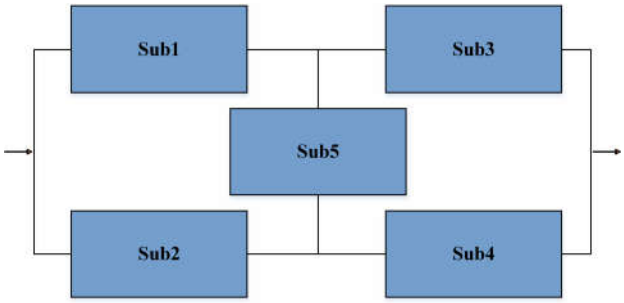
(i) Taking into account the impact of parameters and characteristics of components on the reliability performance of subsystems in RAP.
(ii) Proposing a methodology for modeling the type and extent of dependence among subsystems in RAP.

The rest of the paper is organized as the following. In Section 2 a brief description of bridge system, Copula theory, ANN and PSO is given. In Section 3 the proposed methodology for solving the dependent redundancy allocation problem is illustrated. A numerical example is presented in Section 4 and finally in Section 5 conclusions and suggestions for future research are remarked.

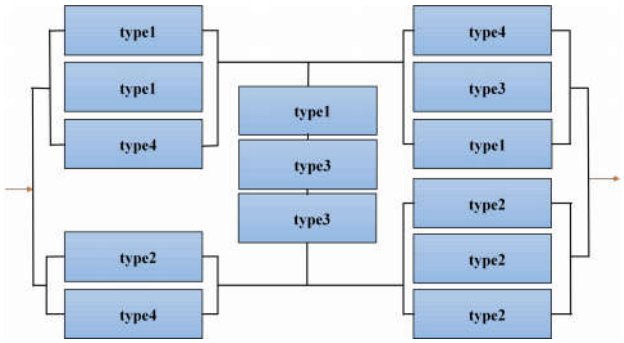
2. Models and methods

2.1. Bridge structure

Bridge topology is a well-known structure commonly used for load balancing and control in various applications such as electric power generation, transmission, computer networks, electronic circuits, etc [24]. Figure 1a Shows a simple bridge structure, which consists of five homogenous subsystems (sub1,...,sub5). To share the imposed load on each subsystem and enhancing the overall reliability of the system, redundant components with different characteristics can be allocated in the subsystems. A redundant bridge structure with nonhomogeneous components is illustrated in Figure 1b. Many researches in the literature have studied



(a) A simple bridge structure



(b) A typical bridge structure with nonhomogeneous redundant components

Figure 1. Bridge topology

the RAP of bridge systems. For more information on this line of research, readers can refer to [1,15,24,25]. The redundancy allocation problem of the bridge system of Figure 1a with constraints on volume, weight, and cost of the system can be formulated as follows:

$$\begin{aligned}\max f(R,N) =& R_1R_2 + R_3R_4 + R_1R_4R_5 \\ &+ R_2R_3R_5 - R_1R_2R_3R_4 - R_1R_2R_3R_5 \\ &- R_1R_2R_4R_5 - R_1R_3R4R_5 - R_2R_3R_4R_5 \\ &+ 2R_1R_2R_3R_4R_5 \end{aligned} \tag{1}$$

subject to:

$$g_1(N) \leq V \tag{2}$$

$$g_2(N) \leq C \tag{3}$$

$$g_3(N) \leq W \tag{4}$$

$$0 \leq r_i \leq 1 \tag{5}$$

$$\mathbf{r_i} \in \mathbb{R} \tag{6}$$

$$i = 1, ..., 5 \tag{7}$$

where, R_i is the reliability of subsystem i for $i = 1, ..., 5$; V, C, and W are the maximum allowed values for volume, cost, and weight of the system, respectively. Also, R and N are, respectively, the reliability and number of components. In addition, $g_j(N)$ for $j = 1, 2, 3$ are functions in terms of the number of components for calculating volume, cost and weight of the system. It should be noted that this formulation is valid for the case of independent subsystems.

2.2. Copula theory

According to Sklar[22], any multivariate joint distribution can be written in terms of univariate marginal distribution functions and a Copula which describes the dependence structure among the variables. Nelsen[17] presented a detailed review of Copula theory and its principles. The Copula is one of the most popular methods for modeling the dependence of data[8], including components life time data. According to Noorossana and Sabri-Laghaie[18], utilizing the Copula in comparison to using traditional multivariate distributions for modeling dependency is very advantageous. Some of the advantages of the Copula method with respect to traditional multivariate distributions are: 1- By the Copula method, one can determine the degree and structure of dependence,2- dependence structure and marginal performance can be specified separately, 3- Copulas are robust to strictly increasing and continuous transformations, 4-univariate marginal functions can be easily derived from different distributions.

Consider H as a joint cumulative distribution function of a vector of continuous random variables $(T_1, ..., T_n)$ with univariate marginals $F_1, ...F_n$. Based on Copula theory a C_θ can be found where:

$$H(T_1, ..., T_n) = C_\theta(F_1(T_1), ..., F_n(T_n)) \quad (8)$$

In which, θ is the vector of Copula parameters expressing dependence among $T_1, ..., T_n$ and can be estimated by means of correlation coefficients.

In order to model the dependence among the lifetimes of components, many Copulas can be used. Choosing an appropriate Copula to model the dependence structure is a critical issue. In reliability problems the dependence among component lifetime is positive and this should be considered in the Copula selection process[8]. One of the most common Copulas used for modeling the dependence of component lifetimes is the Archimedean family. The Archimedean family can be defined as:

$$C(u_1, ..., n_n) = \varphi(\varphi^{-1}(u_1), ..., \varphi^{-1}(u_n)) \quad (9)$$

in which φ is a continuous and non-increasing function $\varphi : [0, \infty] \to [0, 1]$ and is called Archimedean generator. In the present study, Frank, Gumbel and clayton Copulas from the Archimedean family are utilized. Table 1 gives the mathematical definition of these Copulas.

2.3. Artificial neural networks

Artificial neural networks are composed of simple computational elements operating in parallel. These elements are inspired by biological nervous systems. The main applications of ANNs are function approximation by obtaining regression and transformations from input space to feature space by means of nonlinear mapping. An ANN is trained by some data examples to learn the function or transformation mapping and, then, it is used to provide the outputs to the new input data. The ANN has been widely used in various fields, such as pattern recognition[19], classification[16] and prediction[21]. As shown in Figure 2, ANN consists of three main parts: 1-input layer, 2- hidden layer, and 3- output layer. In the input layer, the value of each input is multiplied in by a weight and sent to the nodes of the next layer (called neurons). In the neurons of the hidden layer, a function called activation function is applied to the weighted sum of the inputs to the neuron. The most commonly used activation function is the sigmoid function that is defined as:

$$f(x) = \frac{1}{1 + \exp(-x)} \quad (10)$$

The weights of the output neurons of the network are obtained by applying an activation function, often linear, on the weighted sum of the hidden layer neuron values.

In this study, the ANN is used to model the structure of dependence among the subsystems of bridge system. To do so, it is trained to relate the failure times of the subsystems in the bridge system topology to parameters of the subsystems. In detail, for given input parameters of subsystems, the output will be the type and parameters of the Copula that is suitable for modeling the dependence among the intended subsystems. In this case, the back propagation algorithm is used to train the network. The relationships between the parameters of different subsystems and their corresponding Copula type and parameters are characterized as:

$$CupulaType = F_1(P_1, P_2, ..., P_n) \quad (11)$$

$$\theta = F_2(P_1, P_2, ..., P_n) \quad (12)$$

where $P_1, P_2, ..., P_n$ are vectors of effective parameters of subsystem 1 to subsystem n, θ is the vector of Copula parameters, and F_1 and F_2 are mapping functions from subsystem parameters to Copula type and Copula parameters, respectively.

2.4. Particle Swarm Optimization

Particle Swarm Optimization is a well-known optimization algorithm for the optimization of continuous nonlinear functions, introduced by Kennedy and Eberhart[11]. This algorithm has been inspired by collective behaviors, such as bird flocking and fish schooling. In this method, random solutions, as initial particles, are scattered in solution space and through an iterative procedure, all particles are converged to global optima. In the iterative procedure, the position of each particle is updated by means of its velocity vector, which takes into account the past direction of the particle, best position of the particle in past iterations and best-observed position of all particles in the iterations already elaborated. Then, the position of each particle is updated by its corresponding velocity vector. The mathematical expression of the aforementioned process is stated as follows:

$$v_i^{t+1} = wv_i^t + r_1c_1(pbest_i^t - x_i^t) + r_2c_2(gbest_i^t - x_i^t) \quad (13)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (14)$$

where v_i^{t+1} is the velocity vector of particle i in the iteration t, x_i^t is the position of particle i in the iteration t, and w is an inertia coefficient that expresses the tendency of the particle of keeping its position and takes values between 0 and 1. $pbest_i^t$ and $gbest_i^t$ are the best position of particle$i$

Table 1. Summary of the multivariate Copula functions in this study

Type	Formula	Parameter
Frank	$C_\theta = \frac{-1}{\theta} log\left(1 + \frac{\prod_{i=1}^{d}(e^{-\theta u_i}-1)}{(e^{-\theta}-1)^{d-1}}\right)$	$\theta > 0$
Gumbel	$C_\theta = \exp\left(-\left(\sum_{i=1}^{d}\left(-\log(u_i)\right)^{\theta}\right)^{1/\theta}\right)$	$\theta \geq 1$
Clayton	$C_\theta = \max\left\{\left(\sum_{i=1}^{d}\left(u_i\right)^{-\theta} - (d-1)\right)^{-1/\theta}, 0\right\}$	$\theta \geq \frac{-1}{d-1}, \theta \neq 0$

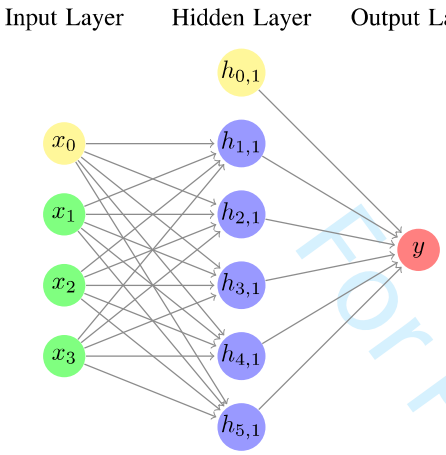


Figure 2. Neural Network structure

in the past iterations and of all particles in the already elaborated iterations, respectively. r_1 and r_2 are random numbers between 0 and 1 and c_1 and c_2 are learning factors, respectively. The steps of this algorithm are as the following:

(i) define algorithm parameters such as number of iterations and population size.
(ii) generate initial population and evaluate the fitness functions.
(iii) update the position of each particle according to Relations 13 and 14 , and then evaluate the fitness function of the new particles.
(iv) stop if the termination condition of the iterative process is met, else go to Step (iii) .

3. Methodology

In this section, a methodology for considering dependence among subsystems in a redundancy allocation problem is proposed with respect to a bridge system configuration. A historical database of subsystem failure times, parameters and configurations of a bridge system are required. In this system, all components work under cold standby strategy and are either operating or failed at any given moment in time. As in the redundancy allocation problem, a combination of components can be used in each subsystem of the system. For the bridge structure, when the subsystems are independent, the reliability of the system can be calculated as Relation 1.

In this study, the Copula theory is utilized to take into account dependence among subsystems. Specifically, an ANN and Copula-based approach are proposed to model the dependence among subsystems. According to this approach,

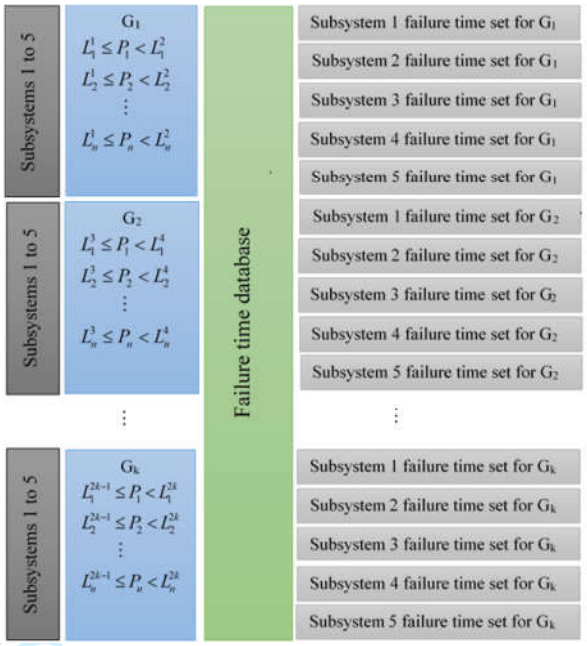


Figure 3. Data gathering process

a relationship between parameters and characteristics of the subsystems and the impact that subsystems may have on each other is established. Subsystems with potentially dependent failure times are chosen and a database of their characteristics and failure history is built. To form this database, parameters and characteristics of components which may be affect the failure times should be included to be considered in the model.

In order to relate parameter values and corresponding failure times, parameter values are classified into specified categories. Each category consists of a combination of parameters with different ranges. By this categorization, each system can be assigned to a specific category according to its parameters. As mentioned, the database contains failure times of systems with different parameter values. Further, the subsystems which have failed and caused the system to fail are recorded in the database. The structure of the database is given in Table 2. In this table, failed subsystem, failure times of subsystems (T), and parameters $(P_1, P_2, ..., P_n)$ of the system are given. Based on this database one can explore the parameter values that may have an effect on the failure of a specific subsystem. Moreover, categories are built for parameters and subsystems according to failed subsystems. This results in a set of parameter values and failure times for each subsystem. Then, the corresponding set of each

Table 2. Database structure

No.	Subsystem 1				Subsystem 2				Subsystem 3				Subsystem 4				Subsystem 5				T	Failed subsystem
	P_1	P_2	...	P_n	P_1	P_2	...	P_n	P_1	P_2	...	P_n	P_1	P_2	...	P_n	P_1	P_2	...	P_n		
1																						
2																						
⋮																						
N																						

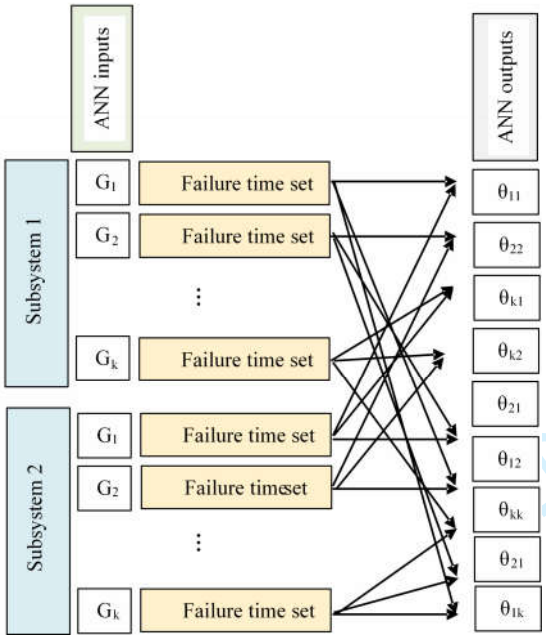


Figure 4. A typical example of ANN inputs and outputs

subsystem is categorized with regard to parameter values in specific ranges. Number and limits of the ranges are chosen according to the database size and domain of the parameters. Next step is to find the set of subsystem failure times corresponding to each parameter's category. Relation between parameter categories and failure time sets is shown in Figure 3. In this figure, G_i for $i = 1, ..., k$ is representative of the ith category of parameters, and $L_j^{(2k-1)}$ and L_j^{2k} for $j = 1, ...n$ are respectively the lower and upper bounds of the jth parameter in category k.

As mentioned before, the reliability of the bridge system in the case when all subsystems are independent is calculated according to Relation 1. When the subsystems are dependent on each other, the reliability function becomes:

$$\begin{aligned} R =& R_{12} + R_{34} + R_{145} \\ &+ R_{235} - R_{1234} - R_{1235} \\ &- R_{1245} - R_{1345} - R_{2345} \\ &+ 2R_{12345} \end{aligned} \tag{15}$$

where, R is the reliability of the system, R_{12}, R_{34}, R_{145}, R_{235}, R_{1234}, R_{1235}, R_{1245}, R_{1345}, R_{2345}, and R_{12345} are, respectively, the joint reliability functions of subsystems (1,2), (3,4), (1,4,5), (2,3,5), (1,2,3,4), (1,2,3,5), (1,2,4,5), (1,3,4,5), (2,3,4,5), and (1,2,3,4,5). In this paper, Copula theory is utilized to model the joint reliability functions of the subsystems. To obtain the Copula model, the following optimisation model is considered:

$$\begin{aligned} \mathbf{max} f(N) =& \mathbf{C}_\theta^{\mathbf{2}}(R_1, R_2) + \mathbf{C}_\theta^{\mathbf{2}}(R_3, R_4) + \mathbf{C}_\theta^{\mathbf{3}}(R_1, R_4, R_5) \\ &+ \mathbf{C}_\theta^{\mathbf{3}}(R_2, R_3, R_5) - \mathbf{C}_\theta^{\mathbf{4}}(R_1, R_2, R_3, R_4) \\ &- \mathbf{C}_\theta^{\mathbf{4}}(R_1, R_2, R_3, R_5) - \mathbf{C}_\theta^{\mathbf{4}}(R_1, R_2, R_4, R_5) \\ &- \mathbf{C}_\theta^{\mathbf{4}}(R_1, R_3, R4, R_5) - \mathbf{C}_\theta^{\mathbf{4}}(R_2, R_3, R_4, R_5) \\ &+ 2\mathbf{C}_\theta^{\mathbf{5}}(R_1, R_2, R_3, R_4, R_5) \end{aligned} \tag{16}$$

subject to:

$$g_1(N) \leq V \tag{17}$$

$$g_2(N) \leq C \tag{18}$$

$$g_3(N) \leq W \tag{19}$$

$$0 \leq r_i \leq 1, \mathbf{r_i} \in real\ number \tag{20}$$

where C_θ^d for $d = 2, 3, 4, 5$ is a d-dimensional Copula function with parameter vector θ for modeling the joint reliability function of the corresponding subsystems. Here, it is supposed that the parameters or characteristics of the subsystems affect the parameter vector of the Copula functions. Therefore, the relation between the parameter vector of the Copula functions and the parameters of the subsystems is obtained. By means of an ANN. The subsystem parameters are the inputs and the Copula parameters are the outputs of the ANN. Suppose that we want to model the joint reliability function between subsystems 1 and 2. To do so, a Copula function is fitted for every combination of parameter categories. Then, an ANN is trained between the fitted Copula parameters and subsystem parameter categories. By means of the trained ANN, one can find the Copula parameters of the joint reliability function between subsystems 1 and 2. This procedure is followed for all combinations of subsystems a joint reliability function is required based on Relation 15. Inputs and outputs of the ANN for building the joint reliability function between subsystems 1 and 2 are depicted in Figure 4, where, θ_{ij} for $i = 1, ..., k$ and $j = 1, ..., k$ is the Copula parameter vector corresponding to the failure time sets of categories i and j.

Three classes of Copulas, Clayton, Gumbel and Frank, are here considered for reliability modeling. Maximum Likelihood Estimation (MLE) method is applied for fitting the Copulas to the failure time data and finding the most appropriate Copula function. A list of ten Copulas should, then, be recorded for every system in the database. Choosing the appropriate Copula class for a new system, a classification process is performed. In order to determine the Copula type, a classifier ANN is trained for each joint

reliability term in Relation 15. The ANN classifier relates subsystem parameters and Copula types. So, according to the values of the subsystem parameters, an appropriate Copula is proposed. Then, as mentioned earlier, an ANN is trained to find the relation between subsystem parameters and the parameter vector of the chosen Copula. By following this procedure one can approximate the type and amount of dependence in a new bridge system just by evaluating the parameters of its subsystems.

A PSO algorithm is utilized to find the optimal configuration in the redundancy allocation problem of the bridge system. During each iteration of the PSO algorithm when solutions are updated to new ones, trained ANNs are applied to find the dependence structure of the new solutions. By knowing the dependence structure, the reliability of the solutions can be calculated based on Relation 16. The procedure of the proposed algorithm is as the following:

(i) Collect data, categorize them and fit the best Copula to failure time data of all required combinations of parameter categories and subsystems.
(ii) Train classifier ANNs for classifying type of Copulas and, then, train ANNs for approximating parameters of the Copulas based on outputs of step 1.
(iii) Generate initial solutions of the ANN algorithm.
(iv) Determine effective parameters of each subsystem and apply trained ANNs to find the dependence structure among subsystems.
(v) Use Relation 16 to calculate the reliability of the solutions (particles).
(vi) Update the position of each particle and calculate fitness of the new particles based on Relation 16.
(vii) Stop if the termination condition is met, else go to Step (vi) .

This procedure is also illustrated in Figure 5.

4. Numerical Example

In order to validate the proposed model, a numerical example is included to show model applicability. As mentioned in Section 3 , all components work under cold standby strategy and are either operating or failed at any moment in time. Also, each subsystem contains nonhomogeneous components. In this regard, a bridge structure with five subsystems is considered where a different number of redundant components can be allocated to each subsystem. Redundant components can be selected from four types of components as, type 1, type 2, type 3 and type 4. All types may fail according to Weibull probability distribution function with parameters as detailed in Table 3. Therefore, reliability of a component at a given time t is:

$$r_i(t) = \exp(\frac{-t}{\alpha_i})^{\gamma_i} \tag{21}$$

where $r_i(t)$ for $i = 1, 2, 3, 4$, is the reliability of component i at time t, and α_i and γ_i are respectively the scale and shape parameters of the ith component Weibull distribution. Three parameters of volume, weight, and cost are recorded for the components. In this regard, volume, weight, and cost of the system should not exceed specified values. It is supposed that parameters P_1 and P_2 are parameters of the components that may affect the reliability performance of other components. For example, P_1 and P_2 can be considered as thermal and radiation coefficients of components. Since there are only constraints on volume, weight, and cost of the system, the maximum allowed values of these parameters are given in Table 3. Also, the maximum number of components in each subsystem is set to be 10. On the other hand, parameters P_1 and P_2 affect the Copula parameters among different subsystems. Hence, the reliability optimization model at a predetermined time (e.g. $t = 100$) can be proposed as follows:

Table 3. Component characteristics

Component type	Weibull parameters		Volume	Weight	Cost
	shape (γ)	scale (α)			
1	0.001	17	40	7	3.5
2	0.02	21	30	5	2
3	0.03	32	30	2	3.5
4	0.004	44	10	1	5

$$\begin{aligned}\mathbf{max} f(N) =& \mathbf{C}^{\mathbf{2}}_{\theta}(R_1, R_2) + \mathbf{C}^{\mathbf{2}}_{\theta}(R_3, R_4) + \mathbf{C}^{\mathbf{3}}_{\theta}(R_1, R_4, R_5) \\ &+ \mathbf{C}^{\mathbf{3}}_{\theta}(R_2, R_3, R_5) - \mathbf{C}^{\mathbf{4}}_{\theta}(R_1, R_2, R_3, R_4) \\ &- \mathbf{C}^{\mathbf{4}}_{\theta}(R_1, R_2, R_3, R_5) - \mathbf{C}^{\mathbf{4}}_{\theta}(R_1, R_2, R_4, R_5) \\ &- \mathbf{C}^{\mathbf{4}}_{\theta}(R_1, R_3, R4, R_5) - \mathbf{C}^{\mathbf{4}}_{\theta}(R_2, R_3, R_4, R_5) \\ &+ 2\mathbf{C}^{\mathbf{5}}_{\theta}(R_1, R_2, R_3, R_4, R_5)\end{aligned} \tag{22}$$

subject to:

$$\sum_{i=1}^{5}\sum_{j=1}^{4} n_{ij} v_j \leq 50 \tag{23}$$

$$\sum_{i=1}^{5}\sum_{j=1}^{4} n_{ij} c_j \leq 25 \tag{24}$$

$$\sum_{i=1}^{5}\sum_{j=1}^{4} n_{ij} w_j \leq 150 \tag{25}$$

$$\sum_{j=1}^{4} n_{ij} \leq 10 \tag{26}$$

$$0 \leq r_i \leq 1, \mathbf{r_i} \in real\ number \tag{27}$$

where, R_i for $i = 1, ..., 5$ is the reliability of subsystem i, r_j, v_j,c_j, and w_j for $j = 1, ..., 4$ are respectively the reliability, volume, cost, and weight of component j. $N = [n_{11}, ..., n_{54}]$ is the vector of component numbers in the subsystems.

Based on the proposed procedure in Section 3 , first a historical database is required. In this regard, a database of failure times and parameter values is generated. A sample of the database and ranges of the parameters are presented in Table 4 and Table 5. According to this database, parameters P_1 and P_2 are used for determining the Copula parameters and volume, cost and weight are constraining parameters. Then, the failure times are categorized based on the failed subsystems. This results in a set of parameter values and failure times for each subsystem. The ranges of parameters are, then, categorized into some sub-ranges. In this example, five sub-ranges are considered for each of the parameters P_1 and P_2. These sub-ranges are given in Table 5. Afterward, the failed subsystems with their corresponding failure times are assigned to the combination of proposed sub-ranges.

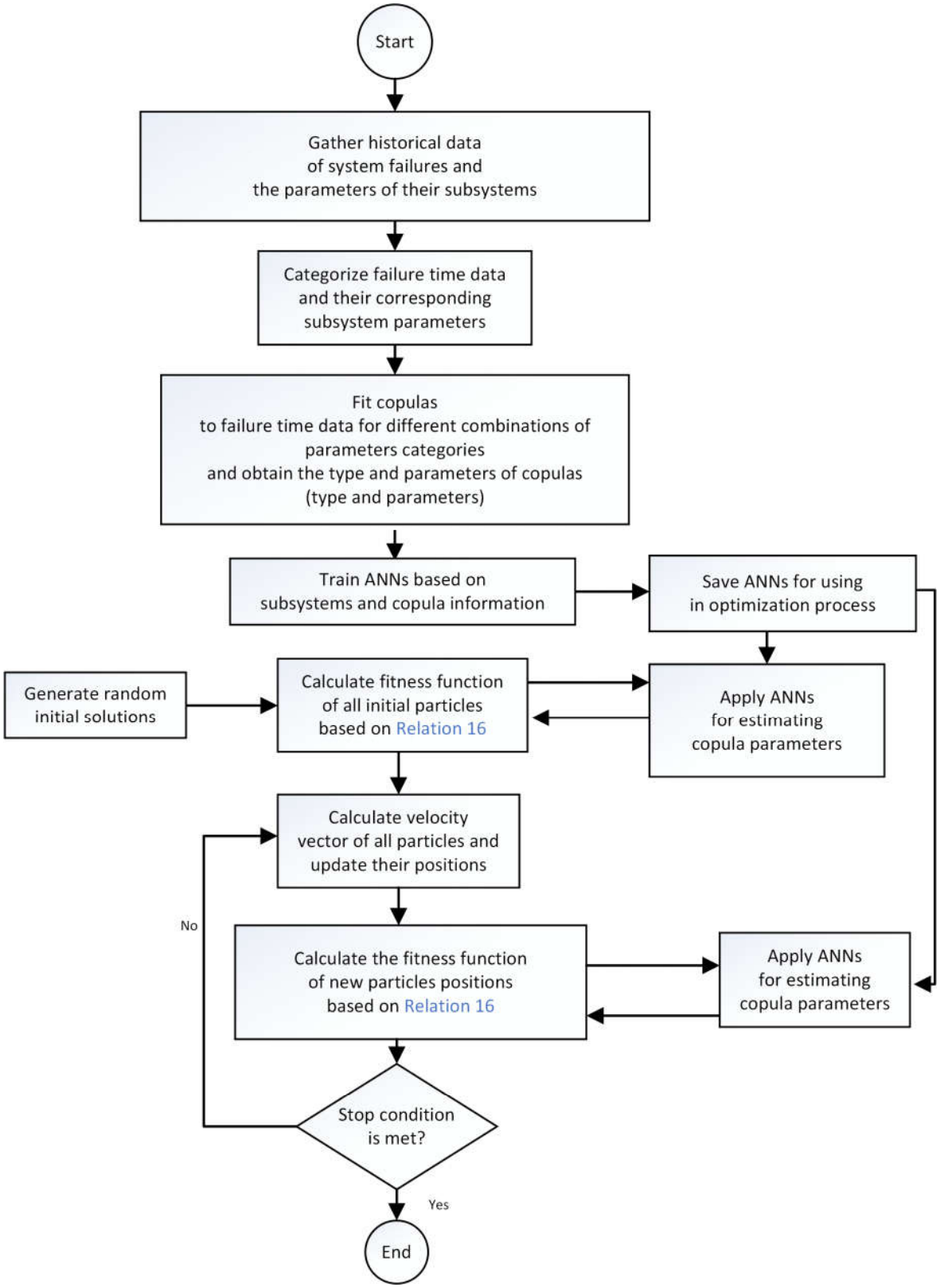


Figure 5. Flowchart of the proposed procedure

This results in 25 sets of failure times for each subsystem, $k = 125$ sets in total. Combinations of sub-ranges which do not contain enough failure times for model fitting can be disregarded. For modeling the joint reliability functions, the most appropriate Copulas among Clayton, Gumbel and frank Copulas are fit to the failure time data. In this regard, for each combination of subsystems in Relation 22 a Copula is required. Therefore, 125 sets of failure times are used for

Table 4. A sample of the database

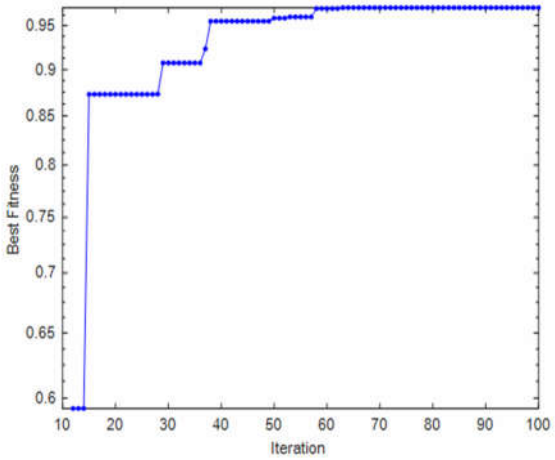
No.	Subsystem 1		Subsystem 2		T	Failed subsystem
	P_1	P_2	P_1	P_2		
1	35	30	15	50	0.4310	1
2	35	15	8	60	0.4876	2
3	7	15	17	120	0.5061	1
4	14	30	27	120	0.5895	2
5	42	180	17	120	0.7988	2
6	42	45	27	90	0.0401	2
7	63	45	180	150	0.0399	1
8	70	50	190	150	0.2974	1
9	14	60	50	60	0.1861	1
10	14	45	45	50	0.2796	1

Table 5. Sub-ranges of parameters (in arbitrary units)

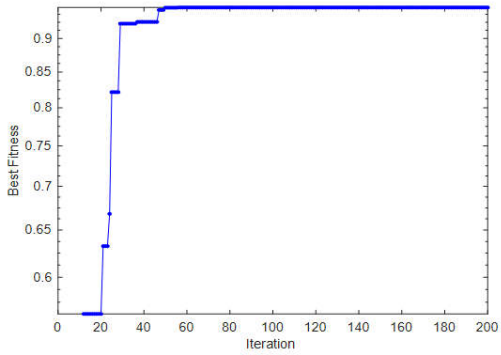
Sub-range number	P_1	P_2
1	1-15	1-40
2	16-30	41-80
3	31-45	81-120
4	46-60	121-160
5	61-75	161-200

Table 6. Structure and performance criteria of classifier ANNs

Subsystems	**Performace criteria (MSE)**			**Hidden layer size**
	train	validation	test	
1,2	0.006	0.007	0.01	[50 50]
3,4	0.05	0.05	0.05	[12 2]
1,4,5	0.01	0.01	0.02	[16 8]
2,3,5	0.03	0.04	0.05	[9 7]
1,2,3,4	0.01	0.02	0.01	[12 10]
1,2,3,5	0.02	0.02	0.02	[9 10]
1,2,4,5	0.03	0.04	0.05	[6 6]
1,3,4,5	0.02	0.04	0.04	[12 5]
2,3,4,5	0.03	0.04	0.05	[9 3]
1,2,3,4,5	0.007	0.004	0.02	[24 12]



(a) Bridge system with independent subsystems



(b) Bridge system with dependent subsystems

Figure 6. Trend of the best fitness versus iteration number of PSO algorithm

determining the type and parameters of the most appropriate Copulas in modeling the joint reliability functions among different combinations of subsystems. As a result, a list of ten Copulas is recorded for every system in the database.

Now, ANNs are trained with input parameter values and output dependence structure (type and parameters of Copulas). Hereby, for every Copula term in Relation 22 a classifier ANN for determining the type of the Copula and an ANN for specifying the Copula parameters are trained. For example, structure and performance criteria (Mean Square Error or MSE) of classifier ANNs trained by the database are represented in Table 6.

After constructing ANNs for classifying the type of Copulas, a similar procedure is performed for modeling the relation between Copula parameters and parameters of subsystems. For example, structure and performance criteria (MSE) of ANNs trained for predicting different Copula parameter values are given in Table 7. All trained ANN structures are saved for estimating the dependence structure of a typical system based on its effective parameter values. For each subsystems combination, the ANN that has the best MSE is chosen for estimating the dependence structure.

After training the ANNs and based on the proposed procedure, the optimum number of redundant components in each subsystem is found by the PSO algorithm. During each iteration of the PSO algorithm and based on Relation 22, the trained ANNs are applied to find the dependence structure of new solutions. In more details, when a new solution is generated during different steps of the PSO algorithm, according to the P_1 and P_2 parameters of that solution and by means of trained ANNs, the type and parameters of the required Copulas are predicted.

The trend of the best fitness versus iteration number of the PSO algorithm in case when subsystems are independent and dependent is represented in Figure 6a and Figure 6b respectively. It can be observed from Figure 6a and Figure 6b that the outputs of the PSO algorithm for dependent subsystems are generally less than the outputs for subsystems that are independent. Also, optimal system configuration for both cases of independent and dependent subsystems is given in Table 8. According to Table 8, reliability of the best configuration when subsystems are dependent is less than the reliability when subsystems are independent.

5. Conclusion

In this paper, the redundancy allocation problem of a bridge system with dependent subsystems was studied.

Table 7. Structure and performance criteria (MSE) of ANNs trained for estimating Copula parameters

Copula types	subsystems	Performace criteria (MSE)			R^2			Hidden layer size
		train	validation	test	train	validation	test	
Clayton		0.003	0.004	0.008	0.99	0.98	0.97	[12 8]
Gumbel	1,2	0.07	0.06	0.07	0.99	0.98	0.98	[4 2]
Frank		0.11	0.12	0.12	0.98	0.98	0.98	[4 2]
Clayton		0.05	0.1	0.1	0.99	0.97	0.89	[14 3]
Gumbel	3,4	0.01	0.04	0.02	0.99	0.98	0.98	[9 4]
Frank		0.02	0.04	0.05	0.99	0.98	0.96	[9 4]
Clayton		0.02	0.07	0.06	0.99	0.98	0.95	[11 3]
Gumbel	1,4,5	0.02	0.03	0.03	0.99	0.99	0.99	[11 2]
Frank		0.03	0.02	0.01	0.99	0.99	0.99	[12 5]
Clayton		0.004	0.007	0.01	0.99	0.97	0.97	[7 6]
Gumbel	2,3,5	0.006	0.01	0.01	0.99	0.99	0.99	[7 6]
Frank		0.007	0.03	0.03	0.99	0.99	0.99	[8 7]
Clayton		0.002	0.02	0.01	0.99	0.98	0.95	[7 6]
Gumbel	1,2,3,4	0.02	0.01	0.02	0.99	0.99	0.98	[7 7]
Frank		0.005	0.02	0.08	0.99	0.99	0.98	[8 9]
Clayton		0.002	0.005	0.04	0.99	0.98	0.98	[11 6]
Gumbel	1,2,3,5	0.006	0.01	0.01	0.99	0.99	0.99	[10 5]
Frank		0.002	0.003	0.003	0.99	0.98	0.97	[11 7
Clayton		0.002	0.003	0.003	0.99	0.98	0.97	[13 5]
Gumbel	1,2,4,5	0.005	0.01	0.02	0.99	0.98	0.98	[10 7]
Frank		0.01	0.1	0.03	0.99	0.98	0.97	[7 5]
Clayton		0.08	0.03	0.03	0.99	0.99	0.98	[10 8]
Gumbel	1,3,4,5	0.002	0.01	0.06	0.99	0.99	0.98	[11 9]
Frank		0.002	0.001	0.003	0.99	0.99	0.99	[17 9]
Clayton		0.002	0.001	0.002	0.99	0.99	0.99	[19 8]
Gumbel	2,3,4,5	0.001	0.003	0.01	0.99	0.98	0.98	[10 7]
Frank		0.01	0.03	0.06	0.99	0.98	0.98	[8 8]
Clayton		0.04	0.05	0.2	0.99	0.98	0.93	[16 5]
Gumbel	1,2,3,4,5	0.001	0.02	0.01	0.99	0.99	0.98	[17 9]
Frank		0.006	0.01	0.02	0.99	0.99	0.98	[12 4]

Table 8. Optimal system configuration

	Subsystem	Type 1	Type 2	Type 3	Type 4	Best fitness
Independent	1	1	0	3	7	0.9637
	2	0	0	0	2	
	3	0	0	9	3	
	4	0	0	0	1	
	5	0	0	0	8	
Dependent	1	0	0	0	5	0.9485
	2	0	0	0	3	
	3	0	4	2	10	
	4	0	0	0	1	
	5	0	0	0	5	

It is supposed that some parameters of components and subsystems can affect the reliability performance of others. In this context, Copula theory was used for modeling the dependence structure among subsystems. The ANNs were applied for modeling the relationship between the parameters of the subsystems and the dependence structure. To do so, a historical database of system parameters and their failure times was used. Then, a particle swarm optimization algorithm was applied for finding the best redundancy structure. Numerical examples show that disregarding dependence can underestimate system reliability.

According to the size of the available database, different approaches can be utilized. The goal of this paper was to propose a methodology which can be useful in modeling dependency in the RAP. In this research, we supposed that enough failure data is available. When enough data for training ANN is not at hand, other approached such as ANFIS or other types of ANNs can be utilized to increase database size or build the predicting model. In addition, experts can be helpful in compensating lack of data. For future research, we aim to propose a methodology which can be useful when enough data is not available.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

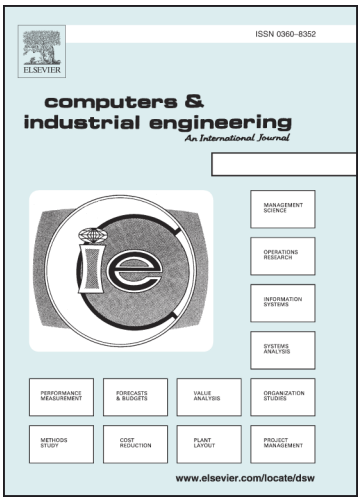
Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

References

1. Mostafa Abouei Ardakan and Mohammad Taghi Rezvan. Multi-objective optimization of reliability-redundancy allocation problem with cold-standby strategy using nsga-ii. *Reliability Engineering & System Safety*, 2017.
2. Noura Beji, Bassem Jarboui, Mansour Eddaly, and Habib Chabchoub. A hybrid particle swarm optimization algorithm for the redundancy allocation problem. *Journal of Computational Science*, 1(3):159–167, 2010.
3. Flix Belzunce, Helena Martnez-Puertas, and Jos M. Ruiz. On optimal allocation of redundant components for series and parallel systems of two dependent components. *Journal of Statistical Planning and Inference*, 141(9):3094 – 3104, 2011. ISSN 0378-3758.
4. Flix Belzunce, Helena Martnez-Puertas, and Jos M. Ruiz. On allocation of redundant components for systems with dependent components. *European Journal of Operational Research*, 230(3):573 – 580, 2013. ISSN 0377-2217.
5. Maw-Sheng Chern. On the computational complexity of reliability redundancy allocation in a series system. *Operations research letters*, 11(5):309–315, 1992.
6. David W Coit. Maximization of system reliability with a choice of redundancy strategies. *IIE transactions*, 35(6):535–543, 2003.

7. Vanderlei da Costa Bueno. Minimal standby redundancy allocation in a k-out-of-n: F system of dependent components. *European Journal of Operational Research*, 165(3):786–793, 2005.
8. Serkan Eryilmaz. Multivariate copula based dynamic reliability modeling with application to weighted-k-out-of-n systems of dependent components. *Structural Safety*, 51:23–28, 2014.
9. Nitin Gupta and Somesh Kumar. Stochastic comparisons of component and system redundancies with dependent components. *Operations Research Letters*, 42(4):284 – 289, 2014. ISSN 0167-6377.
10. Hamideh Jeddi and Mahdi Doostparast. Optimal redundancy allocation problems in engineering systems with dependent component lifetimes. *Applied Stochastic Models in Business and Industry*, 32(2):199–208, 2016.
11. J Kennedy and RC Eberhart. (1995). particle swarm optimization. In *IEEE International Conference on Neural Networks (Perth, Australia), IEEE Service Center, Piscataway, NJ*, pages 1942–1948, 1942.
12. Samuel Kotz, Chin Diew Lai, and Min Xie. On the effect of redundancy for systems with dependent components. *IIE Transactions*, 35(12):1103–1110, 2003.
13. Sadan Kulturel-Konak, Alice E Smith, and David W Coit. Efficiently solving the redundancy allocation problem using tabu search. *IIE transactions*, 35(6):515–526, 2003.
14. Way Kuo and V Rajendra Prasad. An annotated overview of system-reliability optimization. *IEEE Transactions on reliability*, 49(2):176–187, 2000.
15. Chyh-Ming Lai and Wei-Chang Yeh. Two-stage simplified swarm optimization for the redundancy allocation problem in a multi-state bridge system. *Reliability Engineering & System Safety*, 156:148–158, 2016.
16. Yang Lu, Nianyin Zeng, Yurong Liu, and Nan Zhang. A hybrid wavelet neural network and switching particle swarm optimization algorithm for face direction recognition. *Neurocomputing*, 155:219–224, 2015.
17. RB Nelsen. An introduction to copulas, 2nd. *New York: SpringerScience Business Media*, 2006.
18. Rassoul Noorossana and Kamyar Sabri-Laghaie. Reliability and maintenance models for a dependent competing-risk system with multiple time-scales. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, 229(2):131–142, 2015.
19. Amarjit Roy, Salam Shuleenda Devi, and RH Laskar. Impulse noise removal from gray scale images based on ann classification based fuzzy filter. In *Computational Intelligence and Networks (CINE), 2016 2nd International Conference on*, pages 97–101. IEEE, 2016.
20. Seyed Jafar Sadjadi and Roya Soltani. An efficient heuristic versus a robust hybrid meta-heuristic for general framework of serial–parallel redundancy problem. *Reliability Engineering & System Safety*, 94(11):1703–1710, 2009.
21. Nitin Singh, Soumya Ranjan Mohanty, and Rishabh Dev Shukla. Short term electricity price forecast based on environmentally adapted generalized neuron. *Energy*, 125: 127–139, 2017.
22. Abe Sklar. Random variables, distribution functions, and copulas: a personal look backward and forward. *Lecture notes-monograph series*, pages 1–14, 1996.
23. Roya Soltani. Reliability optimization of binary state non-repairable systems: A state of the art survey. *International Journal of Industrial Engineering Computations*, 5(3):339–364, 2014.
24. Yong Wang and Lin Li. A pso algorithm for constrained redundancy allocation in multi-state systems with bridge topology. *Computers & Industrial Engineering*, 68:13–22, 2014.
25. Wei-Chang Yeh, Siang-Tai Wang, Chyh-Ming Lai, Yen-Cheng Huang, Yuk Ying Chung, and Jsen-Shung Lin. Simplified swarm optimization for repairable redundancy allocation problem in multi-state systems with bridge topology. In *Neural Networks (IJCNN), 2016 International Joint Conference on*, pages 3935–3941. IEEE, 2016.
26. Yinping You and Xiaohu Li. On allocating redundancies to koutof n reliability systems. *Applied Stochastic Models in Business and Industry*, 30(3):361–371, 2014.



Solving a Continuous Periodic Review Inventory-Location Allocation Problem in Vendor-Buyer Supply Chain under Uncertainty

Seyed Mohsen Mousavi, Panos M. Pardalos, Seyed Taghi Akhavan Niaki, Armin Fügenschuh, Mahdi Fathi

PII: S0360-8352(18)30674-0
DOI: https://doi.org/10.1016/j.cie.2018.12.071
Reference: CAIE 5624

To appear in: *Computers & Industrial Engineering*

Received Date: 5 July 2018
Revised Date: 11 November 2018
Accepted Date: 29 December 2018

Please cite this article as: Mohsen Mousavi, S., Pardalos, P.M., Taghi Akhavan Niaki, S., Fügenschuh, A., Fathi, M., Solving a Continuous Periodic Review Inventory-Location Allocation Problem in Vendor-Buyer Supply Chain under Uncertainty, *Computers & Industrial Engineering* (2018), doi: https://doi.org/10.1016/j.cie.2018.12.071

Solving a Continuous Periodic Review Inventory-Location Allocation Problem in Vendor-Buyer Supply Chain under Uncertainty

Seyed Mohsen Mousavi[*a]
[a]University of Jyvaskyla, Faculty of Information Technology, P.O. Box 35 (Agora), FI-40014 University of Jyvaskyla, Finland, Email: smousavi@jyu.fi

Panos M. Pardalos[b]
[b]Department of Industrial and Systems Engineering, University of Florida, Gainesville, FL 32611, USA, Email: pardalos@ufl.edu

Seyed Taghi Akhavan Niaki[c]
[c]Department of Industrial Engineering, Sharif University of Technology, P.O. Box 11155-9414 Azadi Ave, Tehran 1458889694 Iran, Email: niaki@sharif.edu

Armin Fügenschuh[d]
[d]Brandenburg University of Technology Cottbus-Senftenberg, Platz der Deutschen Einheit 1, 03046 Cottbus, Germany, Email: fuegenschuh@b-tu.de

Mahdi Fathi[b]
[b]Department of Industrial and Systems Engineering, University of Florida, Gainesville, FL 32611, USA, Email: mfathi.ie@gmail.com

[*] Corresponding author at: University of Jyvaskyla, Faculty of Information Technology, P.O. Box 35 (Agora), FI-40014 University of Jyvaskyla, Finland, e-mail: smousavi@jyu.fi

Solving a Continuous Periodic Review Inventory-Location Allocation Problem in Vendor-Buyer Supply Chain under Uncertainty

Abstract

In this work, a mixed-integer binary non-linear two-echelon inventory problem is formulated for a vendor-buyer supply chain network in which lead times are constant and the demands of buyers follow a normal distribution. In this formulation, the problem is a combination of an (*r, Q*) and periodic review policies based on which an order of size *Q* is placed by a buyer in each fixed period once his/her on hand inventory reaches the reorder point *r* in that period. The constraints are the vendors' warehouse spaces, production restrictions, and total budget. The aim is to find the optimal order quantities of the buyers placed for each vendor in each period alongside the optimal placement of the vendors among the buyers such that the total supply chain cost is minimized. Due to the complexity of the problem, a Modified Genetic Algorithm (MGA) and a Particle Swarm Optimization (PSO) are used to find optimal and near-optimum solutions. In order to assess the quality of the solutions obtained by the algorithms, a mixed integer nonlinear program (MINLP) of the problem is coded in *GAMS*. A design of experiment approach named Taguchi is utilized to adjust the parameters of the algorithms. Finally, a wide range of numerical illustrations is generated and solved to evaluate the performances of the algorithms. The results show that the MGA outperforms the PSO in terms of the fitness function in most of the problems and also is faster than the PSO in terms of CPU time in all the numerical examples.

Keywords*: Inventory-location allocation problem; Mixed-integer binary non-linear programming; Two-echelon supply chain; Stochastic demands; Genetic Algorithm*

1. Introduction

In today's competitive markets companies have to update their logistic systems regularly to capture bigger market share by solving the existing difficulties involved in producing the items, the uncertainties in predicting the demands, the constraints in supplying the items and loading a wide range of items with varying volumes. To reach this aim, the companies need to use preferably the best strategy to integrate their logistic networks as well as their inventory systems, transportation,

warehouses and vendors to minimize the total cost of operations. This research studies a real-world situation of a two-echelon inventory-supply chain problem in which some current limitations in the industry are considered.

Multi-products inventory control problems in finite time-periods have been addressed well by many researchers in recent years. Yang et al. (2017) proposed a mixed-integer linear program for a multi-item inventory problem in finite horizon under non-stationary demand, arbitrary review period, and restricted available inventory budget. Alikar et al. (2017a) modeled a multiple items multiple period inventory control problem for a series-parallel redundancy allocation problem (RAP) in which the total inventory cost was calculated with respect to the time value of money and inflation rates. The total budget for buying the items, the total storage spaces and the truck capacity for transferring the items were limited. Their research was conducted in a deterministic environment with a fixed demand where the lead times were not considered. Alikar et al. (2017b) developed a mixed-integer binary nonlinear model for a multi-product inventory control problem with a finite time-period in a series-parallel RAP problem, in which the products were bought under an all unit discount strategy. In their model, the storage space, the total available budget, the capacities of the vehicles and the system's total weight were constrained. The lead time was assumed to be negligible in their work and also the demands were deterministic. Shankar et al. (2018) presented a mixed-integer nonlinear model for a multiple-product multi-echelon finite horizon inventory-supply chain problem in which some vital factors of the automobile supply chain strategy were integrated. They assumed that no lead times were required. Considering time and cost restrictions, a multi-item multiple periods inventory control problem was improved for a routing model by Peres et al. (2017) where transhipment movements were handled by identical trucks with a unique capacity. They used an exact method and a meta-heuristic algorithm to solve the problem on a case study from a company in the Brazilian retail industry where the demand was assumed fixed and there was no lead time. Liao et al. (2017) proposed a multi-item inventory model in a finite horizon and fuzzy environment with the aim of maximizing the total profits of the retailers. In their work, the lead times were assumed negligible. Mousavi et al. (2013) used a genetic algorithm to optimize an inventory control problem with multiple products in finite time-period where the costs were computed with respect to the time value of money and inflation rates. In their work, discount policies, i.e., an all-unit discount and an incremental quantity discount were applied. The constraints of the problem were the limitation in storage space, supplying order quantity and the total budget at hand. They did not investigate the supply chain members in their work where the demands were deterministic and the lead time was assumed zero. A mixed-integer linear model was developed by Correia & Melo (2017) for a multi-period inventory location-allocation problem, in

which customer segments had different sensitivity to delivery lead times. They used a general-purpose solver to optimize the formulated mixed-integer linear program. The demands in their work were considered deterministic.

In this study, an inventory control problem is formulated for a buyer-vendor supply chain where vendors store their produced items in their own warehouses in order to meet the demands. Supply chain inventory control problem with multiple products and multiple time periods is a popular topic studied by many scholars in different industries. Cárdenas-Barrón et al. (2015) presented a multiple items multi-period inventory lot-sizing problem for a supply chain, in which the best suppliers were to be chosen. To find a near-optimal solution, they solved their problem using an approximation method. No lead times were considered and the demands were deterministic. Sepehri (2011) studied a multiple products multiple time periods inventory model for a supply chain problem where a simulation approach was utilized to solve the problem. The retailers' demands were assumed fixed and there were no lead times for delivery of the products. An inventory control problem with a wide range of items and periods was proposed by Mirzapour Al-e-hashem & Rekik (2014) for a routing problem where items were delivered by capacitated trucks from the suppliers to a plant. Since the model was a mixed-integer linear programming, a standard solver (*IBM ILOG CPLEX*) was used to find the optimal solution of the problem. They modeled the problem with deterministic demands, which can be far from the real world applications. Mousavi et al. (2015) dealt with a multiple products finite horizon inventory-location allocation problem for a retailer-distributer supply chain problem where the distance between retailers and distributors were assumed to be Euclidean and Square Euclidean functions. Two discount strategies as well as all-unit discount and incremental quantity discount were considered and the orders were received in special packets. In their work, a fruit fly optimization algorithm was improved to optimize the proposed problem. Lead times were not considered in their work and the demands were supposed to be deterministic. Moreover, the quality of the solutions found by their applied algorithm was not justified with the one obtained by an exact solution method. A multi-product seasonal (multiple periods) inventory location-allocation problem was formulated by Mousavi et al. (2017b) in a two-echelon buyer-vendor supply chain in which the shortages were not allowed and all-unit discount policy was used to purchase the items. A modified particle-swarm optimization (PSO) algorithm along with a genetic algorithm was utilized to solve the problem. They While the lead times were assumed negligible in their work, they did not assess the performance of their solution algorithms with the one of an exact method. Paksoy & Chang (2010) considered a multi-stage inventory model in a finite horizon for a supply chain problem with multiple popup warehouses and developed a mixed-integer binary linear program. Three multiple

goals were investigated where the revised multi-choice goal programming approach was utilized to solve the problem at hand on a real industrial case study. The customer demands were fixed and no lead times were considered in their research. Jonrinaldi & Zhang (2013) formulated an integrated production multi-item multi-period inventory control problem for a supply chain where several decision making processes and solving methods were used in the proposed mixed integer nonlinear model. Their model assumed constant demand rates and zero lead times.

This article considers an inventory-supply chain problem under uncertainty while the demands of the buyers and the purchasing items from the vendors are stochastic. Rafie-Majd et al. (2018) formulated a three-echelon multi-item multi-period inventory-location problem for a routing supply chain problem where the demands of the customers were considered stochastic. Their approach takes into account the vehicle timetables, fuel consumption, product wastage, and setup cost. Qiu et al. (2017) developed a model for a multi-period inventory control problem structured in a dynamic program with demand uncertainty where a robust optimization method was used to solve the problem. No lead times were investigated in their work. Mousavi et al. (2014) studied an inventory control problem with multiple products in a finite time-period where the total available budget was limited and shortage costs were allowed for all products in combination with backorders and lost sales. They formulated the problem in a fuzzy environment in which the discount rates and the storage space for storing the items were considered as fuzzy numbers. The supply chain members were not brought to the model and the lead times were assumed negligible. Janakiraman et al. (2013) analyzed an inventory control problem in multiple periods for a newsvendor in which the lead times were stochastic and a dilation ordering of lead times implied an ordering of optimal costs. De & Sana (2014) considered a multi-period production-inventory problem with multiple producers in a plant with a multiple shop/delivery system and different machines where the cost function was considered to be fuzzy numbers. Aharon et al. (2009) modeled a multi-period multiple echelons supply chain problem with stochastic uncertainty where a robust optimization method called Affinely Adjustable Robust Counterpart was used to solve the problem. Nasiri et al. (2014) formulated a hierarchical model for designing a production-distribution inventory in a location-allocation problem with multiple-level capacitated warehouses. In order to obtain near-optimal solutions, both Lagrangian relaxation and a genetic algorithm were applied. In order to find better solutions in a shorter time, they employed the Taguchi approach to tune the parameters of their proposed algorithms. In this approach, the number of experiments needed to find the best values of the algorithms' parameters is reduced considerably. There are a number of works published recently in the literature that used the Taguchi approach for tuning the

parameters in inventory and supply chain fields. Interested readers are referred to Mousavi et al. (2015), Mousavi et al. (2017b), Mousavi et al. (2013), and Mousavi et al. (2014) for more details.

The novelties involved in this paper are as follows. First, this work formulates a novel multi-item multi-period inventory-location allocation problem for a two-echelon buyer-vendor supply chain problem. The second novelty is that the problem is formulated under uncertainty while the demands of the buyers are considered stochastic. Moreover, the lead times are assumed constant while it was considered negligible in the related previous works. Furthermore, a modified version of the genetic algorithm, named MGA, and a PSO are applied to obtain near-optimal quantities of the items ordered by the buyers from the vendors in addition to finding near-optimal locations of each vendor placed among the buyers.

The rest of the paper is organized as follows. In the next section, the problem description is given. Indices, notations, and assumptions of the proposed problem come in Section 3. The problem formulations, including the objective function and also the constraints of the model, are presented in Section 4. In Section 5, a modified version of genetic algorithms (MGA) is developed to solve the problem. Section 6 describes the parameter calibration approach and Section 6 shows computational results to evaluate the MGA, in which 20 different numerical examples with different sizes are first generated, and then the Taguchi approach is utilized to tune the algorithm parameters on the generated examples. Finally, the conclusion of the work is described in Section 8.

2. Problem description

In this work, a two-echelon multi-item multi-period inventory control problem is formulated in a buyer-vendor supply chain network, in which the vendors manufacture different products and then store them in their own warehouses to meet the future demands of the buyers. Moreover, the vendors sell their products under an all-unit discount policy, where each vendor can propose different policy with different price break-points. In fact, when a buyer orders a particular item from a vendor, the vendor will charge the buyer based on the quantity of the item requested for which the price break-point provided by the vendor applies. The warehouse spaces, the total budget of the buyers and the total production capacity of the vendors are limited. Furthermore, the vendors deliver their products in special boxes each with a pre-determined number of products. In the model, the demands of the buyers are assumed to be stochastic and all follow a normal distribution where shortages are not allowed. Moreover, lead times of the products are assumed to be constant and there is a limitation on the service levels of the products in each period. The aim is to find out the reorder point in addition to the order quantity of each item so that the total supply

chain cost is minimized. The proposed inventory-supply chain model is shown to be a mixed-integer binary non-linear programming type where two meta-heuristic algorithms, i.e., MGA and PSO, are used to solve the problem. In order to find suitable parameters of the algorithm, a design of experiment approach, i.e. the Taguchi method is used to adjust the MGA and PSO parameters.

Figure 1 shows the supply chain system under investigation. In the next section, the indices, notations, and assumptions of the problem will be presented.

Insert Figure 1 here

3. Indices, notations, and assumptions of the problem

All the notations and indices applied in this work are listed as follows.

3.1. Indices and notations

$i = 1, 2,..., I$ is the index of the buyers

$j = 1, 2,..., J$ is the index of the products

$k = 1, 2,..., K$ is the index of the vendors

$t = 0, 1,..., N$ is the index of the time periods

D_{ijkt}: Expected demand quantity of buyer i for product j produced by vendor k in period t

$f_{ijkt}\left(\mathrm{D}_{ijkt}\right)$: Probability density functions of D_{ijkt} (a normal distribution with mean $\mu_{D_{ijkt}}$ and standard deviation $\delta_{D_{ijkt}}$)

T_{ijkt}: The time at which the j^{th} product ordered by buyer i from vendor k is received

F_k: The production capacity of vendor k

h_{ijkt}: Inventory holding cost per unit of j^{th} product in the warehouse owned by vendor k sold to buyer i in period t

A_{ijkt}: Ordering cost (transportation cost) per unit of j^{th} product from vendor k to buyer i in period t

c_{ijktp}: Purchasing cost per unit of j^{th} product paid by buyer i to vendor k at p^{th} price break point in period t

s_{ijkt}: The required warehouse space for vendor k to store a unit of j^{th} product sold to buyer i in period t

S_i: The available capacity of i^{th} buyer's warehouse

TB: The total available budget

w_{ijkt}: A binary variable that is set to 1 if buyer i orders product j from vendor k in period t, and set to 0 otherwise

Q_{ijkt}: Ordering quantity of j^{th} product purchased by buyer i from vendor k in period t (decision variable)

V_{ijkt}: The number of special boxes of j^{th} product proposed by vendor k to buyer i in period t (decision variable)

n_j: The number of j^{th} product contained in each box

X_{ijkt}: The initial (remained) positive inventory of j^{th} product purchased by buyer i from vendor k in period t (decision variable)

I_{ijkt}: Inventory position j^{th} product for buyer i purchased from vendor k in period t

SS_{ijkt}: Safety stock of j^{th} product for buyer i purchased from vendor k in period t

r_{ijkt}: Reorder point of j^{th} product for buyer i purchased from vendor k in period t

L_{ijkt}: Lead time of j^{th} product for buyer i purchased from vendor k in period t

u_{ijktp}: p^{th} price break-point proposed by vendor k to buyer i for purchasing j^{th} product in period t

λ_{ijktp}: A binary variable that is set to 1 if buyer i purchases product j from vendor k at price break point p in period t, and set to 0 otherwise

$a_i = (a_{i1}, a_{2i})$: The coordinates of the location of buyer i

$y_k = (y_{1k}, y_{2k})$: The potential region of vendor k (decision variable)

M: Maximum inventory level

TC_h: Expected total holding cost

TC_P: Expected total purchasing cost

TC_O: Expected total ordering (transportation) cost

TC: Expected total supply chain cost

3.2. Assumptions

- The buyers' demand rates of all products are stochastic and follow a normal distribution.

- The initial positive inventory level of the items sold out by each vendor to each buyer is zero (i.e., $X_{ijk1} = 0$)
- All orders are placed on a given finite horizon that includes *N fixed* time periods of equal length.
- The orders must be received at the beginning of the next period; thus two scenarios may happen within a period, either the lead time is positive or zero. In other words, if the inventory level reaches below the reorder point, an order is placed and will be received at the beginning of the next period. Even if the inventory level does not reach the reorder point during a period, the order will be received immediately at the beginning of the next period.
- The total storage space, the total production capacity to produce items by each vendor and the total available budget to buy the items are restricted.
- No order is made in the last period.
- The orders arrive in special boxes of a pre-specified number of products.
- The orders should be received at time T, so the lead time would be between the time an order is placed and T.

4. The problem formulation

In this section, we propose a mixed-integer binary non-linear model for the inventory supply chain problem at hand. Figure 2 shows some scenarios of the inventory model.

Insert Figure 2 here

The objective function and the constraints of the model are formulated as follows.

4.1. The objective function

First, let us consider a problem in which shortages are not allowed and the stochastic demands follow a normal distribution. In this problem, the total cost of the proposed supply chain is calculated as:

$$TC = TC_O + TC_h + TC_P . \quad (1)$$

For $\{T_{ijk1}, T_{ijk2}, ..., T_{ijkN}\}; T_{ijkt+1} > T_{ijkt}$ ($for\ t = 1, ..., N$) the total ordering (transportation) cost, the holding cost, and purchasing cost will be obtained as follows.

The total transportation cost is given by Eq. (2).

$$TC_O = \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K}\sum_{t=1}^{N}(Q_{ijkt}A_{ijkt}d(y_k,a_i)), \quad (2)$$

where $d(y_k,a_i)$ is the distance function between the location of vendor *k* and buyer *i*, considering to be the Euclidean function defined as follows:

$$d(y_k,a_i) = \sqrt{(y_{k1}-a_{1i})^2 + (y_{k2}-a_{2i})^2}$$

From Fig 2, the total holding cost will be given by:

$$TC_h = \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K}\sum_{t=1}^{N}\int_{T_{ijkt}}^{T_{ijkt+1}} h_{ijkt}I_{ijkt}dt \quad (3)$$

The demands of the buyers should be covered by the positive level of inventory during lead time L_{ijkt} with a given probability $1-\alpha$ called the inventory service level specified by the decision makers (DMs) where this service level can be formulated as

$$Pr(D_{ijkt} \le r_{ijkt}) = 1-\alpha \quad (4)$$

and we have:

$$r_{ijkt} = \bar{D}_{ijkt} + SS_{ijkt} \quad (5)$$

According to (Miranda & Garrido, 2004), the following formula is the result:

$$r_{ijkt} = E(D_{ijkt}).E(L_{ijkt}) + Z_{1-\alpha}.(\sqrt{(E(D_{ijkt}))^2\delta^2_{L_{ijkt}} + E(L_{ijkt})\delta_{D_{ijkt}}}) \quad (6)$$

Equation (6) is simplified as the following formula when the L_{ijkt} is supposed to be a constant:

$$r_{ijkt} = D_{ijkt}.L_{ijkt} + Z_{1-\alpha}.\sqrt{\delta_{D_{ijkt}}}\sqrt{L_{ijkt}} \quad (7)$$

In Eq. (7), $Z_{1-\alpha}$ is the upper $(1-\alpha)$ percentile point of the standard normal distribution

Figure 2 shows the reorder point situations in the proposed model. Using the Weber problem (Drezner & Hamacher, 2001), the average holding cost rate in the interval period $[T_{ijkt}, T_{ijkt+1}]$ based on the equation above is computed as:

$$TC_h = \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K}\sum_{t=1}^{N-1}\left\{h_{ijkt}\left(\frac{Q_{ijkt}+X_{ijkt}}{2}\right) + h_{ijkt}.Z_{1-\alpha}.\sqrt{L_{ijkt}}\sqrt{\delta_{D_{ijkt}}}\right\} \quad (8)$$

Eq. (8) includes the average cost borne due to storing the order quantity $(Q_{ijkt}+X_{ijkt})$ as the first part which is the inventory level of item *j* applied to cover the buyer demand received during two successive orders. The safety stock is the second average cost included in (8) which is stored in the storage owned by each vendor.

In this work, the vendors sell their products under some discount policies, i.e., all-unit discount and incremental quantity discount. The following equation is the price-break points proposed by the vendors for an all-unit discount policy:

$$\begin{cases} c_{ijkt1} & u_{ijkt1} \le Q_{ijkt} < u_{ijkt2} \\ c_{ijkt2} & u_{ijkt2} \le Q_{ijkt} < u_{ijkt3} \\ & \vdots \\ c_{ijktP} & u_{ijktP} \le Q_{ijkt} \end{cases}$$

Then, the total cost for purchasing the items from the vendor under all-unit discount strategy is calculated as:

$$TC_p = \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K}\sum_{t=1}^{N-1}\sum_{p=1}^{P} Q_{ijkt} c_{ijktp} \lambda_{ijktp} \tag{9}$$

4.2. The constraints

The initial positive inventory of each buyer in each period remained from the previous period is formulated as follows:

$$X_{ijkt+1} = X_{ijkt} + \mathrm{Q}_{ijkt} - \mathrm{D}_{ijkt}(T_{ijkt+1} - T_{ijkt}) \tag{10}$$

Each vendor's warehouse has a limited capacity that is shown by the following equation:

$$\sum_{j=1}^{J}\sum_{k=1}^{N}(Q_{ijkt} + x_{ijkt}) s_{ijkt} \le S_i \tag{11}$$

The products are provided by each vendor in special boxes V_{ijkt} with the number of item n_j where its relevant constraint comes as follows:

$$Q_{ijkt} = n_j V_{ijkt} \tag{12}$$

When the production capacity of each plant owned by each vendor is restricted, the related constraint would be formulated as follows:

$$\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{t=1}^{N} Q_{ijkt} \le F_k \tag{13}$$

The total available budget to buy the products from the vendors is limited which is given by the following formula:

$$\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K}\sum_{t=1}^{N-1}\sum_{p=1}^{P} Q_{ijkt} c_{ijktp} \lambda_{ijktp} \le TB \tag{14}$$

While the order quantity Q_{ijkt} plus the remaining inventory cannot exceed the maximum inventory *M*, the relevant constraint is shown as:

$$Q_{ijkt} + X_{ijkt} \le M \tag{15}$$

Finally, the following constraint describes that a product can be only bought by each buyer at a price break point in each time.

$$\sum_{p=1}^{P} \lambda_{ijktp} = \begin{cases} 1 & if\ \mathrm{Q}_{ijkt} > 0 \\ 0 & otherwise \end{cases} \tag{16}$$

Therefore, the supply chain model for the first model is obtained as follows:

$$MinTC = \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K}\sum_{t=1}^{N}(Q_{ijkt}A_{ijkt}d(y_k, a_i)) + \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K}\sum_{t=1}^{N-1}\{h_{ijkt}.(\frac{Q_{ijkt} + X_{ijkt}}{2}) + h_{ijkt}.Z_{1-\alpha}.\sqrt{L_{ijkt}}\sqrt{\delta_{D_{ijkt}}}\} +$$

$$\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K}\sum_{t=1}^{N-1}\sum_{p=1}^{P} Q_{ijkt}c_{ijktp}\lambda_{ijktp}$$

Subject to:

$$X_{ijkt+1} = X_{ijkt} + \mathrm{Q}_{ijkt} - \mathrm{D}_{ijkt}(T_{ijkt+1} - T_{ijkt})$$

$$\sum_{j=1}^{J}\sum_{k=1}^{K}(Q_{ijkt} + x_{ijkt})s_{ijkt} \le S_i$$

$$Q_{ijkt} = n_j V_{ijkt}$$

$$\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{t=1}^{N} Q_{ijkt} \le F_k$$

$$\sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K}\sum_{t=1}^{N-1}\sum_{p=1}^{P} Q_{ijkt}c_{ijktp}\lambda_{ijktp} \le TB$$

$$Q_{ijkt} + X_{ijkt} \le M$$

$$r_{ijkt} = D_{ijkt}.L_{ijkt} + Z_{1-\alpha}.\sqrt{\delta_{D_{ijkt}}}\sqrt{L_{ijkt}}$$

$$\sum_{p=1}^{P} \lambda_{ijktp} = \begin{cases} 1 & if\ \mathrm{Q}_{ijkt} > 0 \\ 0 & otherwise \end{cases} \tag{17}$$

$$Q_{ijkt} \in Z, x_{ijkt} \ge 0; y_{ijkt}, \lambda_{ijktp} \in \{0,1\};\ (for\ i = 1,2,...,I;\ j = 1,2,...,J;\ k = 1,2,...,K;\ t = 1,2,...,N)$$

5. Solving methodologies

The modified GA and PSO are the solution algorithms used in this paper to solve the problem modeled in (17).

5.1. The MGA

In this research, due to the complexity of the problem a modified version of the genetic algorithm called MGA is used to find out near-optimal order quantities of the products bought from each vendor by each buyer. The MGA steps are described as follows:

- *Initialization of the parameters and representation of the solutions*: The parameters of the MGA are the number of chromosomes (solutions) in the population (*Pop*), the probability of crossover (P_c), the probability of mutation (P_m), and the number of generation (G*en*). The decision variables proposed in this study are *Q* and *y*, where the rest of the decision variables will be obtained, automatically after having *Q* and *y*.
- *Evaluation of the solutions*: In this stage, all the chromosomes of the population are evaluated by the objective function *TC* proposed in Eq. (17). Figure 3 depicts the population of the generated chromosomes evaluated by the *TC* function.

Insert Figure 3 here

- *Selection operator*: After testing several approaches on the problem, a two-chromosome tournament approach is chosen to select two different chromosomes each time randomly and compare them in terms of *TC* after sorting TC of all population solutions in ascending order. The chromosome with the minimum *TC* will be selected to enter the reproduction pool.
- *Crossover operator*: In order to generate new solutions, a crossover operator is performed. First, a number between 0 and 1 is generated randomly for each solution of the population. Then, if the value is less than P_c, the related solution will be chosen for crossover operation. For two different chosen chromosomes R_1 and R_2, the crossover operator is performed using the following formulae:

$$R_1^* = R_1.\mu + R_2(1-\mu)$$

$$R_2^* = R_1.(1-\mu) + R_2\mu \ , \qquad (18)$$

where μ is a random number generated between 0 and 1 and R_1^* and R_2^* are the offspring. Note the value R and R^* include Q and y, where Q is an integer number and y is a number greater than or equal to zero.

- *Mutation operator:* In this paper, a one-point mutation operator is found to be the best approach to generate new solutions for the next generation. First, a random number is

generated between 0 and 1 for each chromosome. If that number is less than P_m, the related chromosome is chosen for the mutation operator. In the chosen chromosome, one gene of Q and two genes of y related to a location are selected randomly and then are changed in the range randomly.

- *Termination criteria:* The algorithm is ended up while the number of generation reaches a pre-specific value (*Gen)*.

5.2. The PSO

In order to validate the results obtained by the proposed MGA, a PSO algorithm is also used to solve the problem. The steps involved in PSO are summarized as follows (Mousavi, et al., 2017a):

- Initializing the parameters and representing the particles the same as shown in Figure 3.
- Initializing the position and velocity of each particle the same as the method performed in (Mousavi et al., 2017a).
- Selecting the process of particles using Pbest and Gbest of each generation.
- Generating new solutions for each particle by updating the positions and velocities.
- Reaching the maximum number of generation as a termination criterion.

6. Experimental design

Tuning parameters in an appropriate way can usually have an impressive effect on the performance of a meta-heuristic algorithm. Since the quality of the solution obtained by any meta-heuristic algorithm such as PSO and GA depends on the values of their chosen parameters, in this section, the Taguchi method is used to tune the parameters. In the work proposed by Eiben & Smit (2011) a conceptual framework for parameter setting in evolutionary algorithms is presented emphasising on two approaches to choose a parameter value: (1) parameter tuning approach, in which the parameter values are set during running the algorithm and (2) the parameter control approach, where the parameter values are changing while running the algorithm. In this work, the first approach is employed.

In a meta-heuristic algorithm, the parameters are controllable factors, the problem being solved is the process input, and the fitness function is the process output. Hence, the best way would be to tune the algorithm's parameters using the experimental design methods as explained as follows instead of applying the values set by other researchers or using a trial and error procedure. In the Taguchi method (Roy, 1990), the factors (here the parameters) which effect on the efficiency (response) of a process are classified into two types: noise factors *N* which are uncontrollable, and

those factors *S* such as the parameters of a meta-heuristic algorithm that are controllable. The Taguchi employs the orthogonal arrays to design the experiments, and then uses an approach to control *N* in order to decrease the variation or scatter around the target; in other words, the design that is impressed less by *N* is a robust design (Sadeghi et al., 2013). In order to analyze the values obtained by the Taguchi, the standard approach and the signal to noise ratio (S/N) approach are utilized. In the standard approach, an analysis of variance is used for experiments with only one iteration whereas the second approach is employed for experiments with more than one iteration. In the meta-heuristic algorithms proposed in this work, more than one replication is needed and thus the second approach has to be applied.

According to *S*/*N* analysis, a good condition is observed if the signal is more than the noise (i.e. $S > N$). In this paper, the aim is to reach a condition that optimizes *S*/*N*. Three categories of characters exist in the Taguchi method, "smaller is better" for which the objective function is of a minimization type, "nominal is the best" for which the objective function has modest variance around its target and "bigger is better", where the objective function is of a maximization type. The *S*/*N* analysis of these three categories is formulated respectively by (Roy, 1990):

$$(S/N)_S = -10 \log\left(\frac{1}{n}\sum_{m=1}^{n} {a_m}^2\right) \quad (19)$$

$$(S/N)_N = -10 \log\left(\frac{1}{n}\sum_{m=1}^{n} (a - a_m)^2\right) \quad (20)$$

$$(S/N)_B = -10 \log\left(\frac{1}{n}\sum_{m=1}^{n} \frac{1}{{a_m}^2}\right), \quad (21)$$

where *n* is the number of iteration, a_m is the response in m^{th} iteration, and *a* is the average response. Using the design of experiment method, i.e. the Taguchi provides the following advantages: (i) reducing the number of iterations, (ii) finding the optimal values of the algorithm parameters and, (iii) reducing the runtime taken by the algorithms to find the best solutions. The implementation of the Taguchi method is explained in the numerical examples in the next section.

7. Computational results and discussions

Some numerical examples are solved in this section in order to demonstrate the application of the proposed methodology as well as to assess the performances of the solution algorithms.

7.1. Numerical examples

As a new type of problem has been addressed in this work, there is no benchmark available in the literature. As such, in this section, 40 numerical examples classified in 20 small-size and 20 large-size problems are generated and solved in order to evaluate the performance of the proposed

solution methods. Then, the Taguchi method is used to obtain the near-optimal values of the MGA parameters on the 40 numerical examples, for which the L_9 array is used. Table 1 shows the input data used to generate the 40 numerical examples with different sizes where the demands of the buyers follow a normal distribution with mean 20 and standard deviation 10 and the other parameters follow a uniform distribution. From Table 1, the coordinates of both the buyers and the vendors are chosen randomly in a region [0, 100]. Tables 2 and 3 depict the 20 small-size numerical examples and their parameter values along with the best and worst results in terms of their fitness values and their required CPU times obtained by the MGA and PSO, respectively. In small-size numerical examples shown in Tables 2 and 3, the number of buyers is between 2 to 15 while this value is between 1 to 10 for the vendors, for the items, and for the time periods while these numbers in large-size numerical examples are 10 to 25 for the buyers, 10 to 20 for the items, 10 to 20 for the vendors and 2 to 3 for the time periods which are shown in Tables 5 and 6. The sixth to ninth columns of Tables 2, 3, 5 and 6 show the optimal levels of the MGA and PSO parameters tuned by the Taguchi method for each numerical problem, respectively. Since the problem is considered as mixed-integer binary nonlinear programming in order to evaluate the quality of the solutions obtained by the MGA and PSO, the problem is coded in *GAMS version 24.1.2* using *MINLP* function. The fitness values and CPU time of small-size numerical examples obtained by GAMS for all 20 small-size numerical examples are shown in Table 4.

In order to clarify how the Taguchi's method works, Problem number 6 (*Prob. No. 6*) of small-size numerical examples is described for the MGA parameters in detail as an example. Table 7 displays the parameters (factors) of the MGA and PSO and their levels which have been found the best values for the generated problems after running the problems many times with different values of the parameters. The Taguchi approach with an L_9 array of the MGA designed for *Prob. No. 6* of small-size numerical examples is shown in Table 8 where the TC value of each combination is brought in the last column. Figure 4 depicts the mean S/N ratio plot for different levels of the parameters for *Prob. No. 6* of small-size numerical examples for the MGA. According to Fig. 4, the best levels of the MGA parameters are $Pop = 200$, $P_C = 0.6$, $P_m = 0.2$ and $Gen = 1000$. In order to show the difference between the best results obtained by the MGA, PSO, and *GAMS* on small-size numerical examples problem, the pictorial representation of the results for TC and CPU time (*hours*) is demonstrated in Figs 5 and 6, respectively. The convergence path of the best results obtained by the MGA for *Prob. No. 6* of small-size numerical examples is shown in Fig 7. Moreover, the obtained optimal orders of the items made by the buyers from the vendors and the optimal locations of the vendors among the buyers resulted by the MGA and GAMS for *Prob. No.*

6 of small-size numerical examples are displayed in Tables 9 and 10, respectively. Figure 8 shows the graphical representation of the optimal locations of the vendors among the buyers for *Prob. No. 6* of small-size numerical examples. In addition, Tables 11 and 12 depict the one-way ANOVA to compare the MGA and PSO for both small-size and large-size numerical examples in terms of the best fitness values and CPU time respectively.

Insert Figures 4 to 8 here

Insert Tables 1 to 12 here

7.2. Discussions

In this section, the results obtained by the proposed methods are analyzed. Since there is no benchmark fit to the model in the literature, 40 different problems are randomly generated and classified into two categories, small-size and large-size, each with 20 numerical examples. This classification is based on the results achieved by the *GAMS version 24.1.2* software and is based on whether the best fitness value can be reached or not running the problem in 6 days continuously. From Tables 2, 3, and 4, the fitness values obtained by the three solution methods are the same for *Prob. No. 1*. However, while the optimal solution is found by all algorithms, the MGA reaches this value faster than the other methods in terms of CPU time (*sec*). In addition, the fitness value obtained by the PSO for *Prob. No. 2* is optimal and is equal to the one achieved by the *GAMS*. Nonetheless, while the solution found by the MGA is not optimal, this algorithm performs better than PSO and *GAMS* in terms of the CPU time. Meanwhile, in *Prob. Nos. 4* and *6*, the MGA reaches the optimal solution in comparison with *GAMS* while it is still the fastest solution method with the lowest CPU time. Moreover, the results in Table 4 show that *GAMS* is not able to solve *Prob. Nos. 14-15* and *17-20* and thus their optimal fitness values are left unknown. In other words, *GAMS* cannot solve the numerical examples of the problems with the number of buyers more than 8, the number of items more than 5, and the number of vendors more than 4 regardless of running the algorithm problem in 6 days conterminously. In fact, the CPU time taken by *GAMS* to solve the numerical examples increases exponentially with the size of the problems which states that the exact methods such as *GAMS* are not suitable for solving the numerical examples of the problem when the dimension of the problem increases.

Comparing MGA with PSO, the results in Tables 2 and 3 are in favor of MGA in terms of the fitness value, except in *Prob. No. 2* where the PSO found a better fitness value. In addition, both algorithms found identical fitness values for *Prob. Nos. 1* and *7*. Furthermore, MGA is the faster algorithm in all the numerical examples solved.

From Tables 5 and 6, the PSO outperforms the MGA in *Prob. Nos. 5, 10, 15, 17* and *19* in terms of fitness value while both algorithms have the same performance to solve *Prob. Nos. 2, 4, 7, 13* and *18*. Of course, the results of fitness values for the rest of the numerical examples are in favor of MGA. The MGA is still faster than PSO in all 20 numerical examples.

To compare the results obtained by both algorithms statistically, the analysis of variance (a one-way ANOVA) is used. Tables 6 and 7 show the one-way ANOVA derived to compare the MGA and the PSO in terms of the fitness value and CPU time for both small-size and large-size numerical examples. According to the p-values shown in these tables, there is no significant difference between the two algorithms in terms of the fitness value and CPU time.

8. Conclusion

In this work, a novel multi-item multi-period inventory-location allocation problem was formulated for a two-echelon buyer-vendor supply chain problem in which the demands of the buyers were considered to be stochastic following a normal distribution. The distance among the buyers and the vendors were assumed to be Euclidean while the available budget, the production capacity, and the storage space to store the items were limited. The objective was to find out the optimal order quantity demanded by the buyers from the vendors and the optimal locations of the vendors among the buyers so that total supply chain cost was as small as possible. While the model was shown to be a mixed-integer binary nonlinear program, the MGA and PSO were used to solve the proposed problem and to find a near-optimum solution. In order to evaluate the quality of the solutions obtained by the algorithms, some small-size numerical examples of the proposed problem were coded and solved by the *GAMS* software. The results showed that with increasing the dimension of the problem, the CPU time taken to solve the problem rose exponentially. The Taguchi's method was also applied to obtain the best parameters value of the algorithms on 40 generated problems of different sizes. The computational results of running both algorithms indicated that the MGA was the better algorithm in most of the numerical examples in terms of the minimum cost and the faster algorithm to solve all problems.

As for recommendations for future, the model can be extended for a routing problem. In addition, the model can be formulated under shortage, inflation and time value of money. Furthermore, some other meta-heuristic algorithms can be used to solve the problem.

References

Aharon, B.-T., Boaz, G. & Shimrit, S. (2009). Robust multi-echelon multi-period inventory control. European Journal of Operational Research, 199(3), 922-935.

Alikar, N., Mousavi, S.M., Ghazilla, R.A.R., Tavana, M. & Olugu, E.U. (2017a). A bi-objective multi-period series-parallel inventory-redundancy allocation problem with time value of money and inflation considerations. Computers & Industrial Engineering, 104, 51-67.

Alikar, N., Mousavi, S.M., Raja Ghazilla, R.A., Tavana, M. & Olugu, E.U. (2017b). Application of the NSGA-II algorithm to a multi-period inventory-redundancy allocation problem in a series-parallel system. Reliability Engineering & System Safety, 160, 1-10.

Cárdenas-Barrón, L.E., González-Velarde, J.L. & Treviño-Garza, G. (2015). A new approach to solve the multi-product multi-period inventory lot sizing with supplier selection problem. Computers & Operations Research, 64(Supplement C), 225-232.

Correia, I. & Melo, T. (2017). A multi-period facility location problem with modular capacity adjustments and flexible demand fulfillment. Computers & Industrial Engineering, 110(Supplement C), 307-321.

De, S.K. & Sana, S.S. (2014). A multi-periods production–inventory model with capacity constraints for multi-manufacturers – A global optimality in intuitionistic fuzzy environment. Applied Mathematics and Computation, 242(Supplement C), 825-841.

Drezner, Z. & Hamacher, H.W. (2001). Facility location: applications and theory: Springer Science & Business Media, Berlin.

Eiben, A.E. & Smit, S.K. (2011). Parameter tuning for configuring and analyzing evolutionary algorithms. Swarm and Evolutionary Computation, 1(1), 19-31.

Janakiraman, G., Park, S.J., Seshadri, S. & Wu, Q. (2013). New results on the newsvendor model and the multi-period inventory model with backordering. Operations Research Letters, 41(4), 373-376.

Jonrinaldi & Zhang, D.Z. (2013). An integrated production and inventory model for a whole manufacturing supply chain involving reverse logistics with finite horizon period. Omega, 41(3), 598-620.

Liao, Z., Leung, S.Y.S., Du, W. & Guo, Z. (2017). A Me-based rough approximation approach for multi-period and multi-product fashion assortment planning problem with substitution. Expert Systems with Applications, 84, 127-142.

Miranda, P.A. & Garrido, R.A. (2004). Incorporating inventory control decisions into a strategic distribution network design model with stochastic demand. Transportation Research Part E: Logistics and Transportation Review, 40(3), 183-207.

Mirzapour Al-e-hashem, S.M.J. & Rekik, Y. (2014). Multi-product multi-period Inventory Routing Problem with a transshipment option: A green approach. International Journal of Production Economics, 157(Supplement C), 80-88.

Mousavi, S.M., Alikar, N., Niaki, S.T.A. & Bahreininejad, A. (2015). Optimizing a location allocation-inventory problem in a two-echelon supply chain network: A modified fruit fly optimization algorithm. Computers & Industrial Engineering, 87, 543-560.

Mousavi, S.M., Alikar, N., Tavana, M. & Di Caprio, D. (2017a). An improved particle swarm optimization model for solving homogeneous discounted series-parallel redundancy allocation problems. Journal of Intelligent Manufacturing. In press, DOI: https://doi.org/10.1007/s10845-017-1311-9.

Mousavi, S.M., Bahreininejad, A., Musa, S.N. & Yusof, F. (2017b). A modified particle swarm optimization for solving the integrated location and inventory control problems in a two-echelon supply chain network. Journal of Intelligent Manufacturing, 28(1), 191-206.

Mousavi, S.M., Hajipour, V., Niaki, S.T.A. & Alikar, N. (2013). Optimizing multi-item multi-period inventory control system with discounted cash flow and inflation: Two calibrated meta-heuristic algorithms. Applied Mathematical Modelling, 37(4), 2241-2256.

Mousavi, S.M., Sadeghi, J., Niaki, S.T.A., Alikar, N., Bahreininejad, A. & Metselaar, H.S.C. (2014). Two parameter-tuned meta-heuristics for a discounted inventory control problem in a fuzzy environment. Information Sciences, 276, 42-62.

Nasiri, G.R., Zolfaghari, R. & Davoudpour, H. (2014). An integrated supply chain production-distribution planning with stochastic demands. Computers & Industrial Engineering, 77(Supplement C), 35-45.

Paksoy, T. & Chang, C.-T. (2010). Revised multi-choice goal programming for multi-period, multi-stage inventory controlled supply chain model with popup stores in Guerrilla marketing. Applied Mathematical Modelling, 34(11), 3586-3598.

Peres, I.T., Repolho, H.M., Martinelli, R. & Monteiro, N.J. (2017). Optimization in inventory-routing problem with planned transshipment: A case study in the retail industry. International Journal of Production Economics, 193, 748-756.

Qiu, R., Sun, M. & Lim, Y.F. (2017). Optimizing (s, S) policies for multi-period inventory models with demand distribution uncertainty: Robust dynamic programing approaches. European Journal of Operational Research, 261(3), 880-892.

Rafie-Majd, Z., Pasandideh, S.H.R. & Naderi, B. (2018). Modelling and solving the integrated inventory-location-routing problem in a multi-period and multi-perishable product supply chain with uncertainty: Lagrangian relaxation algorithm. Computers & chemical engineering, 109(Supplement C), 9-22.

Roy, R. A primer on the Taguchi method, Society of Manufacturing Engineers, Michigan, 1990.

Sadeghi, J., Mousavi, S.M., Niaki, S.T.A. & Sadeghi, S. (2013). Optimizing a multi-vendor multi-retailer vendor managed inventory problem: Two tuned meta-heuristic algorithms. Knowledge-Based Systems, 50, 159-170.

Sepehri, M. (2011). Cost and inventory benefits of cooperation in multi-period and multi-product supply. Scientia Iranica, 18(3), 731-741.

Shankar, R., Bhattacharyya, S. & Choudhary, A. (2018). A decision model for a strategic closed-loop supply chain to reclaim End-of-Life Vehicles. International Journal of Production Economics, 195, 273-286.

Yang, L., Li, H., Campbell, J.F. & Sweeney, D.C. (2017). Integrated multi-period dynamic inventory classification and control. International Journal of Production Economics, 189(Supplement C), 86-96.

The Tables

Table 1. The input data for generating the numerical problems

Parameters	Distribution function
D	$N(20,10)$
F	$U(50000,1000000)$
h	$U(3,20)$
A	$U(5,20)$
c	$U(10,20)$
s	$U(1,10)$
S	$U(1000000,5000000)$
TB	$U(1000000,10000000)$
n	$U(2,6)$
u	$U(0,50)$
a	$U(0,100)$
y	$U(0,100)$
M	$U(0,150)$
μ	$U(20,50)$
σ	$U(10,15)$

Table 2. The general data for different small-size numerical examples along with the fitness function and CPU time of the MGA

Prob. No.	Number of Buyers	Number of Items	Number of Vendors	Number of Time periods	MGA							
					Pop	Pc	Pm	Gen	Fitness		CPU time (Sec)	
									Best	Worst	Best	Worst
1	2	2	1	2	50	0.6	0.2	500	**1.919e^{4}**	2.102e^{4}	2.75	2.95
2	2	2	2	2	50	0.6	0.2	500	2.212e^{4}	2.745e^{4}	1.58	1.83
3	3	2	2	2	50	0.7	0.1	500	3.129e^{4}	3.986e^{4}	1.13	1.45
4	4	3	2	2	50	0.6	0.2	500	**2.090e^{5}**	2.432e^{5}	7.21	10.49
5	4	4	2	2	100	0.6	0.2	500	3.033e^{5}	3.477e^{5}	17.61	21.21
6	5	2	2	2	200	0.6	0.2	1000	**8.793e^{4}**	1.139e^{5}	12.74	13.18
7	5	4	3	3	100	0.6	0.2	500	6.724e^{5}	7.771e^{5}	22.31	23.11
8	5	5	3	3	200	0.6	0.2	500	9.146e^{5}	1.222e^{6}	28.56	29.42
9	5	5	4	5	200	0.6	0.2	500	3.608e^{6}	3.714e^{6}	52.99	55.77
10	8	2	2	2	100	0.6	0.2	500	2.618e^{5}	3.110e^{6}	23.12	24.905
11	8	3	3	3	100	0.6	0.2	500	3.393e^{6}	2.441e^{6}	26.39	28.58
12	8	4	4	4	100	0.6	0.2	500	6.080e^{6}	6.218e^{6}	36.42	38.41
13	8	5	4	4	200	0.6	0.2	500	7.379e^{6}	7.650e^{6}	83.25	86.42
14	8	5	5	5	200	0.6	0.2	1000	8.690e^{6}	8.803e^{6}	240.76	248.60
15	8	6	6	6	200	0.6	0.2	1000	2.397e^{7}	2.409e^{7}	303.81	310.02
16	10	2	2	2	200	0.6	0.2	500	3.778e^{5}	4.175e^{5}	54.54	56.71
17	10	4	4	4	200	0.6	0.2	500	7.074e^{6}	7.275e^{6}	69.08	72.14
18	10	8	5	5	200	0.7	0.2	1000	1.814e^{7}	1.826e^{7}	459.48	464.53
19	10	8	8	8	200	0.7	0.2	1000	7.856e^{7}	7.901e^{7}	992.94	998.23
20	15	10	10	10	200	0.8	0.1	1000	2.162e^{8}	2.315e^{8}	1085.16	1098.75

Table 3. The general data for different small-size numerical examples along with the fitness function and CPU time of the PSO

Prob. No.	Number of Buyers	Number of Items	Number of Vendors	Number of Time periods	PSO							
					C_1	C_2	Pop	Gen	Fitness		CPU time (Sec)	
									Best	Worst	Best	Worst
1	2	2	1	2	1.5	2	70	700	**1.919e^4**	2.131e^4	2.83	3.19
2	2	2	2	2	2	1.5	100	700	**2.205e^4**	2.890e^4	1.63	1.89
3	3	2	2	2	1.5	2.5	100	700	3.163e^4	4.222e^4	1.32	1.46
4	4	3	2	2	2	1.5	200	1000	2.136e^5	2.583e^5	7.51	11.60
5	4	4	2	2	2	1.5	200	1000	3.209e^5	3.524e^5	18.08	22.43
6	5	2	2	2	2	2.5	200	1200	8.901e^4	1.540e^5	12.95	13.88
7	5	4	3	3	2	1.5	100	700	6.724e^5	7.997e^5	24.78	25.91
8	5	5	3	3	2.5	1.5	200	1000	9.381e^5	1.420e^6	30.24	32.62
9	5	5	4	5	2	1.5	200	700	3.611e^6	3.783e^6	55.47	57.32
10	8	2	2	2	2	2.5	100	700	2.685e^5	3.222e^6	23.45	27.20
11	8	3	3	3	2	1.5	200	1000	3.396e^6	2.521e^6	27.74	31.43
12	8	4	4	4	1.5	1.5	100	1200	6.200e^6	6.821e^6	37.66	42.23
13	8	5	4	4	2	2	100	700	7.411e^6	8.005e^6	86.98	91.32
14	8	5	5	5	2	1.5	200	1000	8.719e^6	8.851e^6	248.28	256.72
15	8	6	6	6	1.5	1.5	200	1200	2.405e^7	2.430e^7	311.46	319.56
16	10	2	2	2	2	2.5	200	700	3.796e^5	4.272e^5	57.33	59.21
17	10	4	4	4	2	2	200	1000	7.144e^6	7.327e^6	72.31	76.84
18	10	8	5	5	2.5	2	100	1200	1.823e^7	1.872e^7	468.92	476.21
19	10	8	8	8	2	1.5	200	1200	7.898e^7	8.051e^7	1005.38	1034.28
20	15	10	10	10	2	2.5	200	1000	2.173e^8	2.386e^8	1104.20	1118.39

Table 4. The general data for different small-size numerical examples and the fitness function and CPU time obtained by GAMS

Prob. No.	Number of Buyers	Number of Items	Number of Vendors	Number of Time periods	GAMS	
					Fitness	CPU time (Sec)
1	2	2	1	2	**1.919e^{4}**	26.31
2	2	2	2	2	**2.205e^{4}**	43.72
3	3	2	2	2	3.125e^{4}	59.26
4	4	3	2	2	**2.090e^{5}**	213.45
5	4	4	2	2	3.029e^{5}	418.75
6	5	2	2	2	**8.793e^{4}**	421.96
7	5	4	3	3	6.716e^{5}	1022.75
8	5	5	3	3	9.138e^{5}	7653.44
9	5	5	4	5	3.590e^{6}	24536.52
10	8	2	2	2	2.604e^{5}	18782.35
11	8	3	3	3	3.382e^{6}	108369.39
12	8	4	4	4	6.066e^{6}	232136.31
13	8	5	4	4	7.359e^{6}	475183.49
14	8	5	5	5	-	-
15	8	6	6	6	-	-
16	10	2	2	2	3.758e^{5}	432540.23
17	10	4	4	4	-	-
18	10	8	5	5	-	-
19	10	8	8	8	-	-
20	15	10	10	10	-	-

Table 5. The general data for different large-size numerical examples along with the fitness function and CPU time of the MGA

Prob. No.	Number of Buyers	Number of Items	Number of Vendors	Number of Time periods	MGA							
					Pop	Pc	Pm	Gen	Fitness		CPU time (Sec)	
									Best	Worst	Best	Worst
1	10	10	10	2	100	0.7	0.2	500	$3.723e^{6}$	$4.130e^{6}$	292.11	354.30
2	10	15	10	2	100	0.7	0.1	500	$4.826e^{6}$	$5.103e^{6}$	558.97	631.21
3	10	15	13	2	100	0.6	0.2	1000	$5.165e^{6}$	$5.596e^{6}$	610.20	676.22
4	10	10	15	2	100	0.7	0.1	500	$5.004e^{6}$	$5.496e^{6}$	588.53	690.13
5	15	10	10	2	100	0.7	0.1	500	$6.171e^{6}$	$6.587e^{6}$	692.98	741.27
6	15	15	10	2	100	0.6	0.2	1200	$7.381e^{6}$	$7.823e^{6}$	802.03	859.08
7	15	10	15	3	200	0.6	0.2	1000	$7.237e^{6}$	$7.892e^{6}$	851.26	932.38
8	15	15	12	3	100	0.7	0.2	1200	$8.401e^{6}$	$8.923e^{6}$	1000.39	1201.36
9	15	15	14	3	200	0.7	0.1	1000	$8.900e^{6}$	$9.310e^{6}$	1031.93	1201.58
10	15	15	15	2	100	0.7	0.1	1000	$9.803e^{6}$	$1.060e^{7}$	1307.38	1443.61
11	10	15	15	3	100	0.7	0.2	1200	$9.119e^{6}$	$9.831e^{6}$	1281.32	1399.02
12	17	15	10	3	100	0.7	0.2	1200	$1.989e^{7}$	$2.197e^{7}$	1532.24	1675.28
13	20	10	10	3	200	0.6	0.2	1000	$1.123e^{7}$	$1.238e^{7}$	1885.25	2100.63
14	20	10	15	2	200	0.6	0.2	1000	$3.230e^{7}$	$3.402e^{7}$	2799.00	3320.45
15	20	15	10	2	100	0.7	0.1	1200	$5.128e^{7}$	$5.652e^{7}$	2895.32	3579.65
16	20	15	15	2	200	0.7	0.2	1200	$8.220e^{7}$	$8.959e^{7}$	4010.25	4950.38
17	20	20	10	2	200	0.7	0.2	1200	$7.456e^{7}$	$7.838e^{7}$	4302.46	5181.33
18	20	20	15	2	200	0.8	0.1	1200	$8.233e^{7}$	$8.881e^{7}$	4831.92	5920.64
19	20	20	20	2	200	0.8	0.2	1200	$9.010e^{7}$	$9.263e^{7}$	5098.52	6672.20
20	25	20	15	2	200	0.7	0.2	1200	$2.230e^{8}$	$2.432e^{8}$	7002.28	9543.37

Table 6. The general data for different large-size numerical examples along with the fitness function and CPU time of the PSO

Prob. No.	Number of Buyers	Number of Items	Number of Vendors	Number of Time periods	PSO							
					C_1	C_2	Pop	Gen	Fitness		CPU time (Sec)	
									Best	Worst	Best	Worst
1	10	10	10	2	2	2	100	1000	$3.811e^{6}$	$4.231e^{6}$	312.23	398.23
2	10	15	10	2	1.5	2	70	1200	$4.826e^{6}$	$5.132e^{6}$	591.21	603.28
3	10	15	13	2	1.5	1.5	100	1000	$5.273e^{6}$	$5.641e^{6}$	645.32	691.65
4	10	10	15	2	2	2	100	1000	$5.004e^{6}$	$5.412e^{6}$	627.50	711.35
5	15	10	10	2	2.5	1.5	100	1000	$6.160e^{6}$	$6.616e^{6}$	718.22	769.18
6	15	15	10	2	1.5	2	200	1000	$7.409e^{6}$	$7.900e^{6}$	822.33	871.53
7	15	10	15	3	2	2	100	1200	$7.237e^{6}$	$7.856e^{6}$	903.12	1012.01
8	15	15	12	3	1.5	1.5	200	1000	$8.383e^{6}$	$8.955e^{6}$	1032.22	1152.86
9	15	15	14	3	1.5	1.5	200	1000	$8.919e^{6}$	$9.341e^{6}$	1142.40	1225.15
10	15	15	15	2	2	1.5	200	700	$9.720e^{6}$	$1.022e^{7}$	1343.21	1401.21
11	10	15	15	3	2	1.5	100	1200	$9.122e^{6}$	$9.815e^{6}$	1327.11	1410.22
12	17	15	10	3	2	2	100	1000	$2.021e^{7}$	$2.371e^{7}$	1622.24	1731.06
13	20	10	10	3	1.5	1.5	200	1200	$1.123e^{7}$	$1.220e^{7}$	1956.20	2190.30
14	20	10	15	2	2	2	200	1000	$3.235e^{7}$	$3.418e^{7}$	2986.18	3572.71
15	20	15	10	2	1.5	2	100	1200	$5.125e^{7}$	$5.654e^{7}$	3012.51	3809.60
16	20	15	15	2	1.5	1.5	200	1000	$8.238e^{7}$	$8.962e^{7}$	4200.20	4969.33
17	20	20	10	2	2	2	200	1000	$7.453e^{7}$	$7.831e^{7}$	4272.22	5109.33
18	20	20	15	2	1.5	2.5	200	1200	$8.233e^{7}$	$8.870e^{7}$	4901.90	5996.38
19	20	20	20	2	2	2	200	1200	$9.003e^{7}$	$9.251e^{7}$	5325.02	7030.29
20	25	20	15	2	1.5	2.5	200	1200	$2.235e^{8}$	$2.400e^{8}$	7112.20	9918.37

Table 7. The parameters levels of the algorithms

Algorithm	Parameter	Low (1)	Medium (2)	High (3)
MGA	Pop	50	100	200
	Pc	0.5	0.6	0.7
	Pm	0.1	0.15	0.2
	Gen	200	500	1000
PSO	C1	1.5	2	2.5
	C2	1.5	2	2.5
	Pop	70	100	200
	Gen	700	1000	1200

Table 8. The Taguchi array of the MGA for *Prob. No. 6* of small-size numerical examples

Array	Pop	Pc	Pm	Gen	TC
1	1	1	1	1	883241
2	1	2	2	2	881250
3	1	3	3	3	880568
4	2	1	2	3	879985
5	2	2	3	1	881456
6	2	3	1	2	881425
7	3	1	3	2	879985
8	3	2	1	3	880365
9	3	3	2	1	881135

Table 9. The optimal orders made by the buyers from the vendors obtained by the MGA and GAMS for *Prob. No. 6* of small-size numerical examples

	Two types of item produced by Vendor 1 in the first period		Two types of item produced by Vendor 2 in the first period	
Buyer	1	2	1	2
1	100	29	102	93
2	34	74	86	75
3	30	54	100	76
4	65	130	29	94
5	55	25	89	105

Table 10. The optimal location of the vendors among the buyers obtained by the MGA and GAMS for *Prob. No. 6* of small-size numerical examples

Optimal location of Vendor 1		Optimal location of Vendor 2	
y_{11}	y_{12}	y_{21}	y_{22}
8	12	13	26

Table 11. The one-way ANOVA to compare MGA and PSO for small-size numerical examples in terms of the best fitness values and CPU time

	Source	DF	Adj SS	Adj MS	F-Value	P-Value
Fitness value	Factor	1	1.00255E+11	1.00255E+11	0.00	0.995
	Error	38	9.44628E+16	2.48586E+15		
	Total	39	9.44629E+16			
CPU time	Factor	1	147	147	0.00	0.970
	Error	38	3911720	102940		
	Total	39	3911867			

Table 12. The one-way ANOVA to compare MGA and PSO for large-size numerical examples in terms of the best fitness values and CPU time

	Source	DF	Adj SS	Adj MS	F-Value	P-Value
Fitness value	Factor	1	27772900000	27772900000	0.00	0.998
	Error	38	1.08420E+17	2.85316E+15		
	Total	39	1.08420E+17			
CPU time	Factor	1	54701	54701	0.01	0.904
	Error	38	141322528	3719014		
	Total	39	141377228			

The Figures

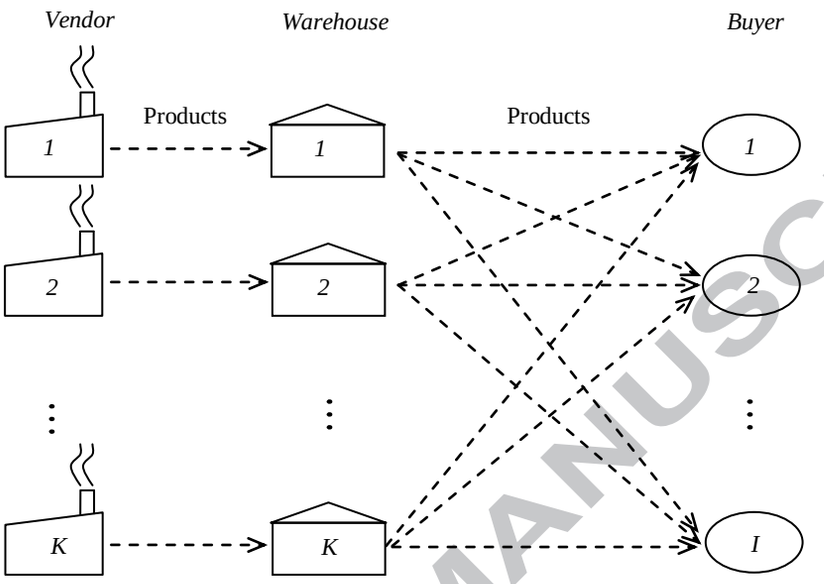


Fig 1. The supply chain system

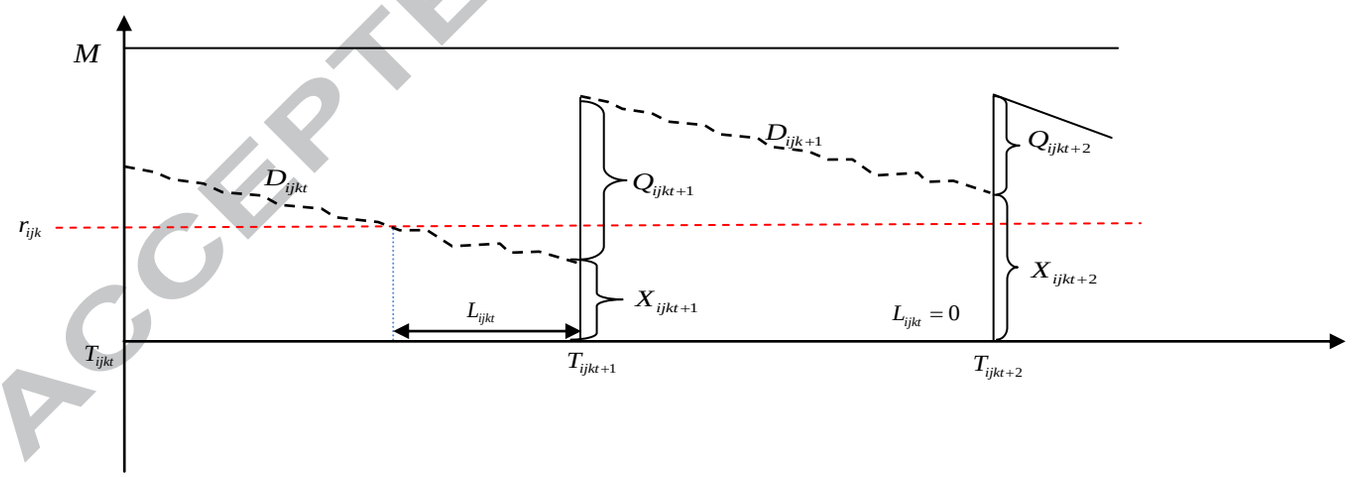


Fig 2. Two different scenarios of net stock vs. time for the inventory model

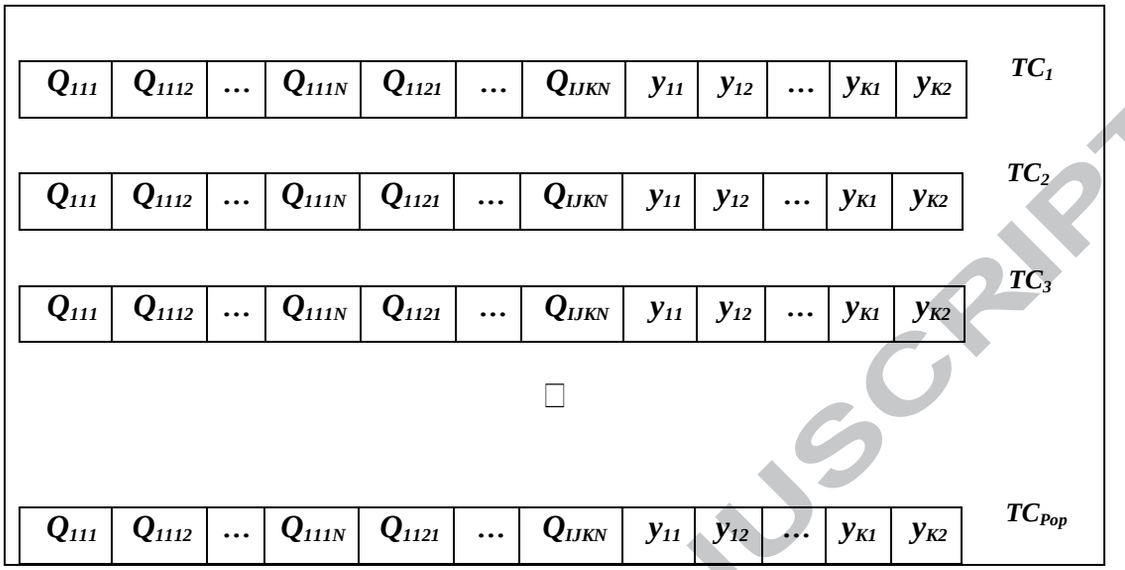


Fig 3. The representation of a chromosome

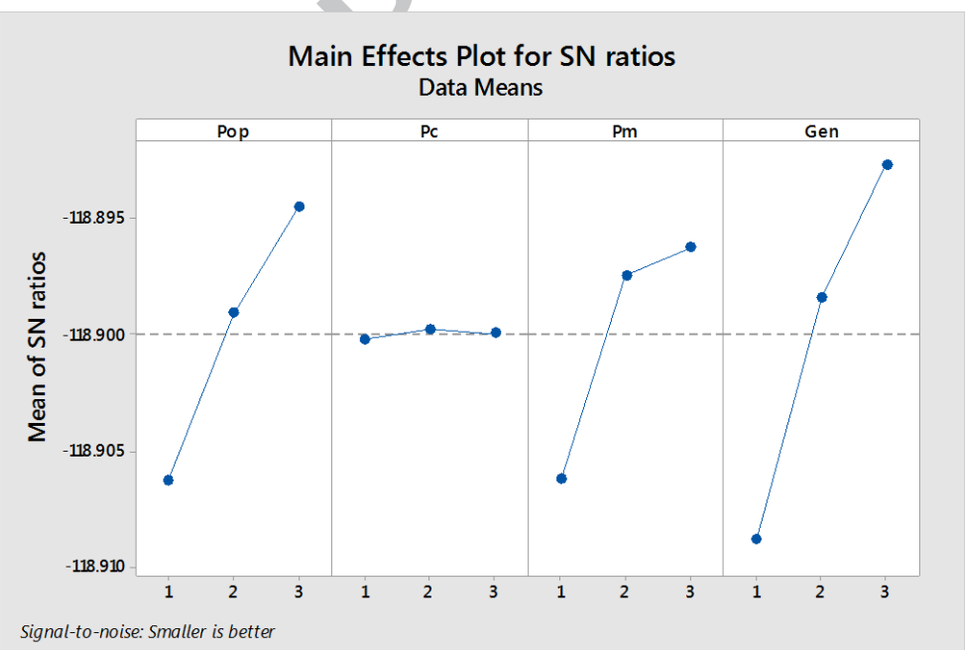


Fig 4. The mean S/N ratio plot for different levels of the parameters for *Prob. No. 6* of small-size numerical examples for the MGA

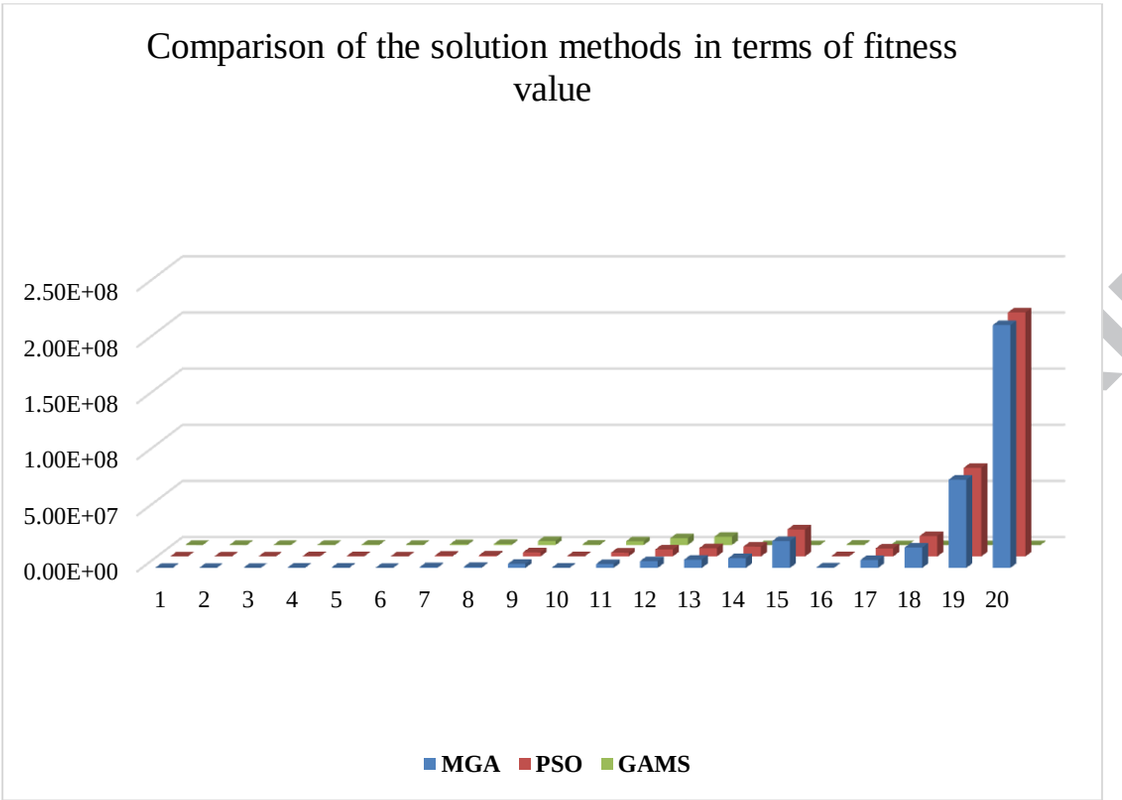


Fig 5. The gap between the best and the worst results of TC obtained by the MGA, PSO and GAMS for small-size numerical examples

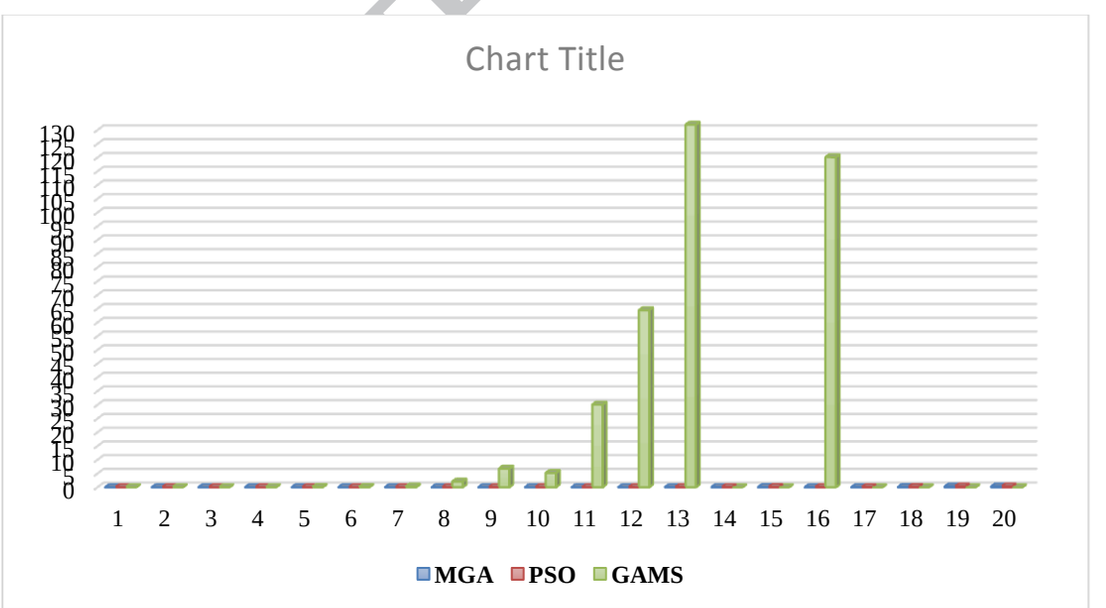


Fig 6. The gap between the best and the worst results of CPU time (*hours*) obtained by the MGA, PSO, and GAMS for small-size numerical examples

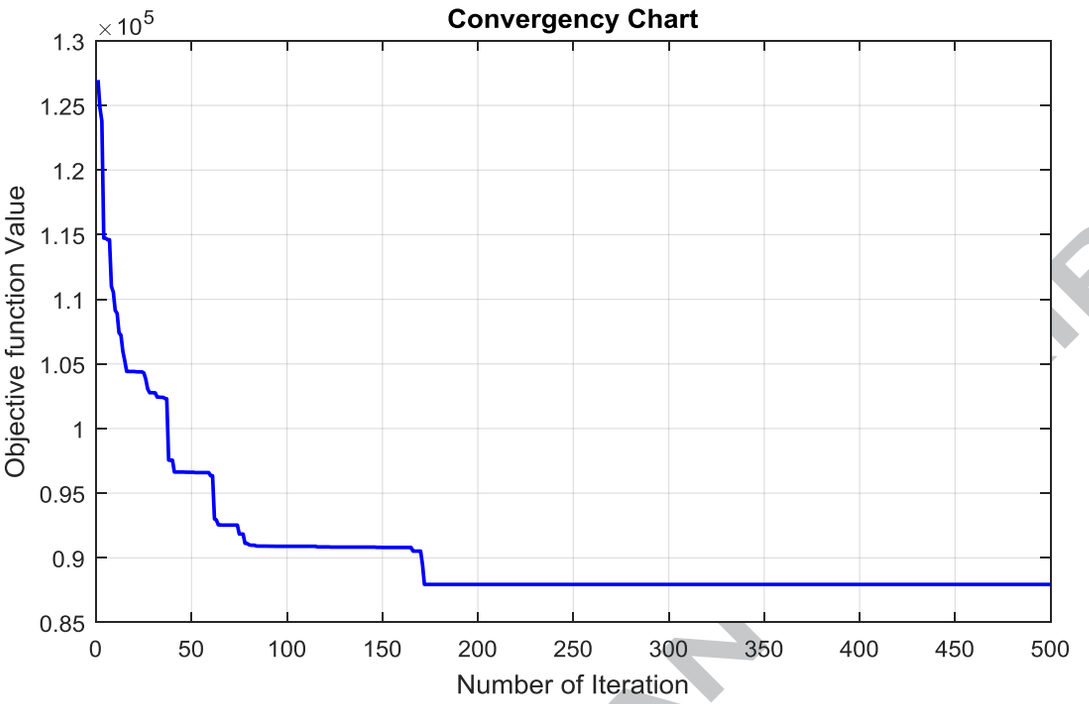


Fig 7. The convergence path of the best results obtained by the MGA for *Prob. No. 6* of small-size data

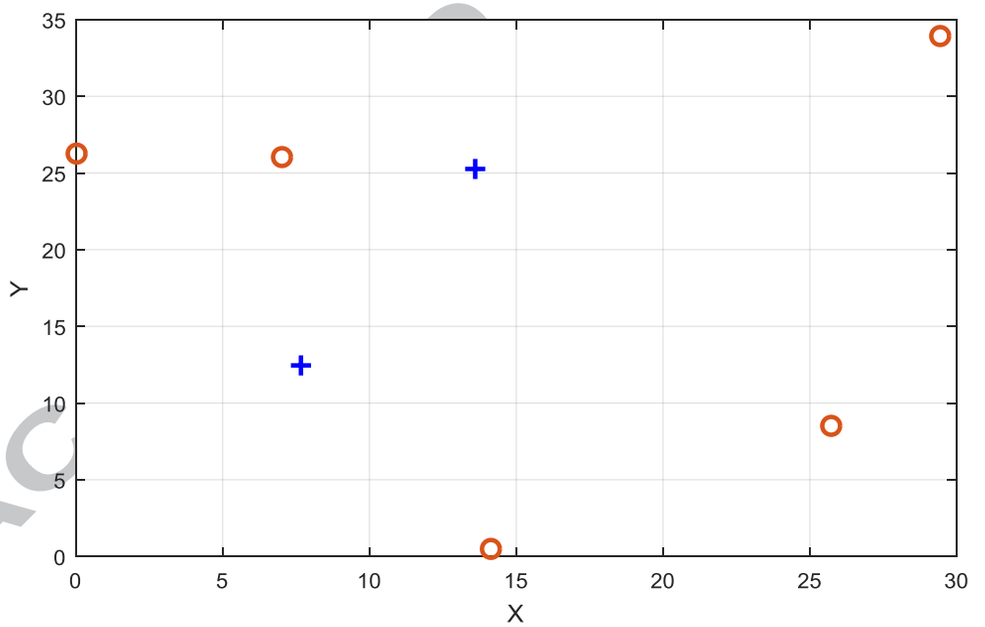


Fig 8. The optimal location of the vendors (blue points) among the buyers (orange circles) obtained by the MGA and GAMS for *Prob. No. 6* of small-size data

Highlights

1. A mixed-integer binary non-linear two-echelon stochastic inventory problem is formulated where the demands of buyers are stochastic.
2. The problem is formulated to be a combination of an (r,Q) and periodic review policies
3. The aim is to find the optimal order quantities and the optimal placement of the vendors such that the costs are minimized.
4. A Genetic Algorithm and Particle Swarm Optimization are used.
5. A design of experiment approach is utilized to adjust the parameters of the algorithms.

Compensating Misalignment Using Dynamic Random-Effect Control System: A Case of High-Mixed Wafer Fabrication

Marzieh Khakifirooz, *Student Member, IEEE*, Chen-Fu Chien, *Member, IEEE*, and Mahdi Fathi, *Member, IEEE*

***Abstract*—It is vital to have an exclusive modification in semiconductor production process because of meeting differentiated customer demands in dynamic and competitive global minuscule semiconductor technology market and the highly complex fabrication process. In this paper, we propose a control system based on the dynamic mixed-effect least-square support vector regression (LS-SVR) control system for overlay error compensation with stochastic metrology delay to minimize the misalignment of the patterning process. Moreover, for the stability of the control system in the presence of metrology delay and to deal with nonlinearity among the overlay factors, the novel Lyapunov-based kernel function is merged with the LS-SVR controller. The proposed controller's operation has been validated and implemented by a major semiconductor manufacturer in Taiwan. The experiments are verified that mixed-effect LS-SVR controller has the higher validity and higher efficiency in comparison with the exponentially weighted moving average (EWMA) and double EWMA controllers which had been previously implemented at the company or applied in similar studies.**

***Note to Practitioners*—Due to high production complexity, a meticulous and intelligent process control is needed to achieve higher throughput and customer satisfaction. Monitoring a complex system is challenging because the process components and variables operate autonomously and interoperate with other manufacturing segments. This paper proposes a novel run-to-run control system to compensate for the overlay error during the photolithography process that efficiently deals with the high-mixed manufacturing environment and metrology delay.**

***Index Terms*—High-mixed process, Lyapunov stability, metrology delay, overlay error, photolithography process, recipe-based system, support vector regression (SVR).**

AQ:1 Manuscript received May 6, 2018; accepted January 13, 2019. This paper was recommended for publication by Associate Editor Z. Yin and AQ:2 Editor XXXXXX upon evaluation of the reviewers' comments. This work was supported by the Ministry of Science and Technology, Taiwan, under Grant MOST 107-2634-F-007-002 and Grant MOST 107-2634-F-007-009. *(Corresponding author: Chen-Fu Chien.)*

AQ:3 M. Khakifirooz and C.-F. Chien are with the Department of Industrial AQ:4 Engineering and Engineering Management, National Tsing Hua University, Hsinchu, Taiwan, and also with the Artificial Intelligence for Intelligent Manufacturing Systems Research Center, Ministry of Science and Technology, Taipei 10622, Taiwan (e-mail: khakifirooz.marzieh@gapp.nthu.edu.tw; cfchien@mx.nthu.edu.tw).

M. Fathi is with the Department of Industrial and Systems Engineering, Mississippi State University, Starkville, MS 39762 USA (e-mail: fathi@ise.msstate.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASE.2019.2894668

I. Introduction

SEMICONDUCTOR devices are continuing to shrink in size so that process engineers and researchers are facing these issues daily that how they can adopt a more authentic monitoring system to get them rid of receiving intensive out of control errors and enhance the yield [1]. More critically, the lithography process deals with entire new difficulties attached to development in postlithographic technologies. Consequently, the factory integration team is required to investigate challenges to ensure infrastructure readiness for the lithography process and improve the advanced process control (APC) with the tighten control limits. Therefore, the overlay error of the lithography process is selected for further investigation.

Run-to-run (R2R) control has been extensively adapted to analyze a variety of challenges in the process monitoring of complex semiconductor manufacturing (see [2], [3]). For more details, one can refer to comprehensive reviews of studies on APC methods (the Kalman filtering technique [4], a dynamic dead band controller [5], stochastic sequential AQ:5 optimization [6], artificial neural network method [7], and feedforward-feedback learning-based controller [8]).

Misalignment in photolithography process is studied in several papers [9]–[11]. Also, several papers considered mathematical modeling for overlay alignment [12]–[14]. Additional prior reviews of overlay error compensation in photolithography process include [15]–[18]. A research of literature revealed several studies that design a proper APC system for high-mixed semiconductor plant such as threaded double exponentially weighted moving average (dEWMA) [19], a combined product and tool disturbance estimator [20], and a cycle forecasting EWMA [21] controller.

Due to the need for the provision of rapid feedback to the process control, the lack of real-time metrology data causes extensive limitations in the R2R control. Most semiconductor manufacturing processes suffer from issues caused by metrology delays due to the time needed for measurements, metrology capacity, and the waiting time in the wafer queue between the processing tool and the metrology station [22]. The stability and performance of the process will be affected by the metrology delay. Moreover, since quality measurements perform online, the delay would not be fixed but flows stochastically.

On the other hand, tuning of control parameters quickly and optimally is extensively required to achieve an acceptable control performance for intelligent manufacturing of modern fabs. However, most of the control models cannot update themselves, as the dynamics of the system vary during online control, and thus it may endure from modeling inaccuracies. As a function approximator should minimize the total risk, yet most approximators such as neural networks and polynomial estimators minimize empirical risks. The limited training set, compared to the number of free parameters, can cause a high generalized risk of overfitting. By minimizing the empirical risk, in combination with generalized risk, a more efficient approximation technique for reducing the total risk called structural risk minimization (SRM) [23] can be obtained. The SRM technique implements the support vector regression (SVR). Both the optimal control problems and SVR methods are a type of optimization models. Hence, one could try to merge these two formulations.

In this paper, we aim to introduce the multiple-input single-output (MISO) controller based on least-square SVR (LS-SVR) to minimizing the unmeasurable disturbance affected by the stochastic delay from metrology tools to fabrication tools and process noise. The LS-SVR controller [24] has gained popularity due to its promising performance in minimizing the regret function. Several studies have adopted the SVR method for monitoring dynamic multiple nodes process [25]. The contributions of proposed LS-SVR method are: 1) to set up the tuned parameters of the LS-SVR controller of the high-mixed recipe system; 2) to deal with the unmeasurable delay and disturbance during the lithography process; 3) to compensate the misalignment of overlay factors in the high-mixed environment; and 4) to investigate the stability of the system in the presence of stochastic delay and deal with nonlinearity among the overlay factors.

The remainder of this paper is organized as follows: Section II introduces the MISO system framework for overlay factors. Section III presents the recipe-based control system in semiconductor manufacturing with a discussion of properties of random effect LS-SVR controller including the Lyapunov-based polynomial-kernel function. Section IV demonstrates the simulation experiments for a high-mixed plant and analyzes the sensitivity of the proposed LS-SVR controller. Section V includes analysis of manufacturing data. Finally, we conclude this paper in Section VI.

II. Fundamentals

A. Notation and Terminologies

The notation and terminologies used in this paper are listed as follows.

i	Recipe index.
j	Overlay factor index.
t	Process run index.
k	Fold index for cross validation.
N	Number of overlay factors in the system.
K	Number of fold for cross validation.
c_j	Indicator for controller of overlay factor j.
p_j	Indicator for plant of overlay factor j.
R_i	Indicator for recipe i.
u_{tj}	Input variable for overlay factor j at run t.
Q_{tj}	Process output for overlay factor j at run t.
d_{tj}	Process disturbance for overlay factor j at run t.
E_{tj}	Deviation from the target for overlay factor j at run t.
z_{tj}	Random effect for overlay factor j at run t.
T	Target of overlay factors.
ε_t	White noise in a process for run t.
τ	Process gain (parameter of EWMA/dEWMA controller).
a	Parameter of EWMA/dEWMA controller.
θ	Fixed discount factor in EWMA/dEWMA controller.
D_t	Drift parameter of dEWMA controller at run t.
$\mathbf{x}_{tj}$	State vector of overlay factor in state-space model for overlay factor j at run t.
$C(.)$	LS-SVR cost function from optimization.
$\mathbb{C}$	LS-SVR regularization parameter.
b	LS-SVR bias term.
$\kappa(.,.)$	LS-SVR kernel function.
$(\boldsymbol{\alpha} - \boldsymbol{\alpha}*)$	LS-SVR support vector.
$u_{t(j)}$	Input variable for jth overlay factor with highest E_t at run t.
$Q_{t(j)}$	Process output for jth overlay factor with highest E_t at run t.
$C_{t(j)}(.)$	Cost function for for jth overlay factor with highest E_t at run t.
R_{emp}	Empirical risk function of control system.
$f, g, \mathbf{h}$	Mapping functions in stat-space model.
δ_{t_i}	Stochastic drift for recipe i at run t.
M	Upper bound for total overlay error.
σ	Admissibility parameter.
η	LS-SVR online learning parameter.
V_{Lyap}	Lyapunov stability function.
$\mathbf{P}$, $\mathbf{P}^*$	Lyapunov positive definite symmetric matrices.
λ, γ	Parameters of Zero-Inflated Poison (ZIP) distribution.
$\mathbf{I}$	Identity matrix/vector.
m, n	Intercept and power parameters for polynomial kernel function.
$\boldsymbol{\mu}$	Mean vector.
$\boldsymbol{\Sigma}$	Variance covariance matrix.

B. Multiple-Input Single-Output Control System

A system in which multiple inputs are used to govern a single output is called a MISO system. Regards to the communication status among input variables, the MISO system can carry noncollaborative and collaborative control strategies. The noncollaborative control plant is equivalent to a single-input-single-output system with a single feedback loop for the most critical input variable [26]. On the other hand, the collaborative system, either in serial or parallel structure [27], benefits from the dynamic status of each plant to improve the performance of monitoring system considering the restrictions of process inputs and individual outputs. This paper designs a customized serial collaborative MISO controller in which

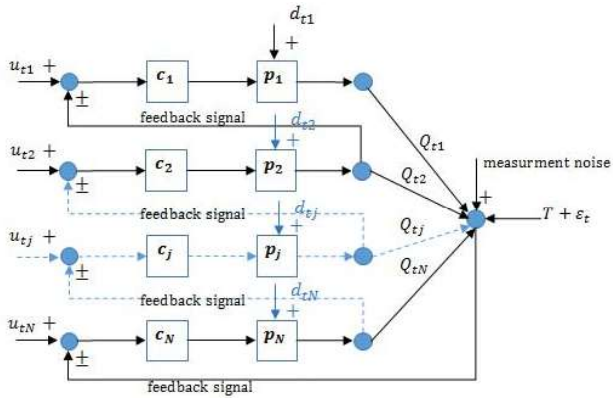


Fig. 1. MISO control system for overlay error compensation.

overlay factors, relations, and attitudes are adopted. A regular linear ten-factor overlay model used by [17] is contemplated in this paper. Regarding the model in [17], although ten factors were investigated to compensate for the overlay error, the target point for all factors remains the same. Therefore, the MISO model is well suited for our study considering the collaboration and dependence among all overlay variables.

C. MISO Controller for Overlay Factors

Assume an R2R-MISO system where plants are arranged decreasingly based on the deviation of controller outputs and the target point of the last run. The procedure of a cascade control system [28] is well used for controlling such dynamic control system where the system can be dynamic as a mix of linear or nonlinear plants. However, the linearity or nonlinearity remains fix at the entire process. Generally, a cascade control system consists of at least two control loops, at least one inner loop and one outer loop for closed-loop system operation. Load disturbance that enforces into the inner loop(s) can prevent before it extends to the whole system (outer loop). Therefore, if the inner loop contains a significant disturbance, the system can react and compensate the disturbance faster before the outer loop run or the whole system will be affected.

Consider Fig. 1, with N input variables, each serial plant p_j, $j = 1, \ldots, N$ at tth run, defines the effect of each manipulated variable u_{tj} over the corresponding estimated output $(\widehat{Q}_{tj})$. The nonmeasurable disturbance d_{tj} deviates each output variables from their corresponding control plant p_j. A set of serial controllers c_j are designed to minimize such deviation. Each c_j is updated based on a lower level measurement output Q_{tj+1}, which is corrupted by the noise signal. The level or order of each serial plant and controller is dynamic at $(t+1)$th run according to the error function in (1) at tth run, in a way that overlay factors with higher deviations have priority to monitor at the next run

$$\mathbf{E}_t = \mathbf{Q}_t - \mathbf{I}(T + \varepsilon_t) \tag{1}$$

where $\mathbf{I}$ is a identity vector.

Algorithm 1: R2R-MISO

for $l : 1 \rightarrow t$ **do**
  Receive $(\boldsymbol{u}_l + d_l)$ for N overlay factors;
  for $j : 1 \rightarrow N$ **do**
    Create $(u_{l(j)} + d_l)$;
    Estimate $\widehat{Q}_{l(j)}$;
    Receive $(T + \varepsilon_l)$;
    Update $c_{(j)}$ controller by $\widehat{Q}_{l(j+1)})$;
    if $\left|C_{(j)} - C_{(j+1)}\right| \leq \varepsilon_l$ **then**
      $\widehat{Q}_{l(j+1)}, \ldots, \widehat{Q}_{l(N)} = \widehat{Q}_{l-1(j+1)}, \ldots, \widehat{Q}_{l-1(N)}$;
      Break loop;
    end
  end
end
Randomly split the data into K disjoint set;
if $\forall k : 1 \rightarrow K : \Pr\left(\sup\left|C_{(j)k} - C_{(j+1)k'}\right| \leq \sigma\right) \geq 1 - \eta$ **then**
  Return $\widehat{u}_{l(j)}$, $\widehat{Q}_{l(j)}$ and $C_{(j)}$
end

D. Research Framework for the Proposed R2R-MISO Control System

Algorithm 1 presents the general operational structure of the proposed dynamic monitoring system in this paper. The initial analysis criteria of the presented control system are consisting of: 1) optimization analysis; 2) providing the cost and constraints of the optimization problem; 3) the multiple input nodes estimation; and 4) stabilization of the estimated results.

Several other phenomena including the delay of information flowing from the metrology tool to the control plant, and being a bottleneck make severe disturbances. Furthermore, at the bottleneck, multiple recipes are associated with the single machine. This situation can be more complicated when multiple products are manufactured during a specific time, and the production line schedule is a mixed schedule. To arrange this complication, a straightforward mixed system as a guideline for advanced high-mixed fab has investigated.

III. Recipe-Based Control System for Semiconductor Manufacturing

This section introduces the mixed-recipe control system based on the LS-SVR controller and the nonlinearity among the overlay factors is also considered as another challenging issue. Assume the recipe-based controller based on a mixed-recipe system. Disregarding generalities, considering different levels of complexity as shown in Figs. 2–4. Fig. 2 shows a particular case of a mixed-recipe schedule where each recipe is periodically applied to the process after a certain number of runs, δ, (fixed drift). Within a fixed drift recipe schedule system, the subsequent run's control action is based on the previous run when the same recipe was on that tool. If the recipe schedule is random (stochastic drift, δ_t) (Fig. 3), the subsequent run's control action is controlled based on the output of the previous run, when the same recipe and drift were

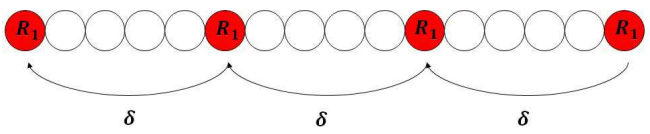


Fig. 2. Mixed-recipe schedule system with fixed drift.

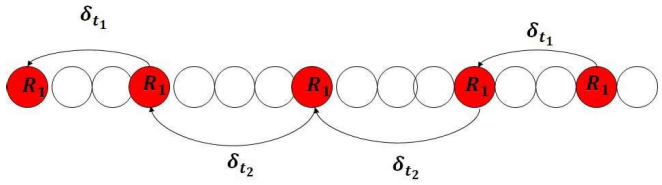


Fig. 3. Mixed-recipe schedule system with random drift.

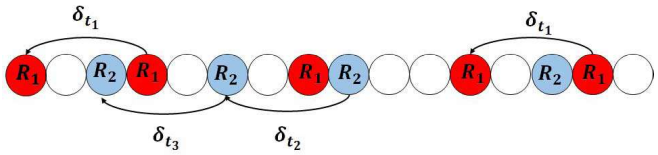


Fig. 4. Mixed-recipe schedule system with random drift and random sequence.

on that tool. The system becomes even more complicated with a dynamic schedule and in the presence of other recipes in the system, as shown in Fig. 4.

In this paper, we consider a system with high-mixed recipe schedule with a random drift similar to Fig. 4. The random drift recipe schedule shown in Fig. 4 considers as the random intercept or random offset into the linear or nonlinear regression model. Therefore, to optimize the performance of the controller, a random effect LS-SVR algorithm has been applied when the polynomial kernel function based on Lyapunov stability condition is employed as a mapping function to enhance the performance of the proposed controller.

A. Review of LS-SVR

The dynamic model proposed in this paper compensates for the majority of process dynamics and noise-based disturbances, such as gain and offsets of multivariate processes [29], [30], the quadratic effects of process variables [31], autocorrelation and deterministic drifting effects [32], stochastic metrology delay [33], and nonstationary disturbances [34]. For these purposes, consider a set of training points $\{(\mathbf{u}_1+\mathbf{d}_1, T+\varepsilon_1), \ldots, (\mathbf{u}_{t-1}+\mathbf{d}_{t-1}, T+\varepsilon_{t-1})\}$. The model used to carry out the optimization and identification of the empirical risk of approximation as follows:

$$\min\ R_{\text{emp}} = \frac{1}{t}\sum_{l=1}^{t} C_l(T+\varepsilon_l, \widehat{\mathbf{Q}}_l)$$
$$\text{s.t. } \widehat{\mathbf{Q}}_t = \mathbf{h}(\widehat{\mathbf{Q}}_{t-1}, \ldots, \widehat{\mathbf{Q}}_1, \mathbf{u}_{t-1}, \ldots, \mathbf{u}_1, \mathbf{d}_{t-1}) \quad (2)$$

where both $C_t(.)$ and $\mathbf{h}(.,.)$ are assumed to be twice continuously differentiable. This model structure is intended for modeling a dynamic system with input $\mathbf{u}_t \in R^n$ and output $\mathbf{Q}_t \in R^n$.

The control objective is designed to provide the control signal, based on the system and an adaptation law, for adjusting control parameters. Therefore, the state vector of the approximator function in the presence of an external disturbance follows the desired trajectory state (Target), and the tracking error in (1), converges to zero, when $\varepsilon_t \sim N(0, \sigma_\varepsilon)$ and $E(\varepsilon_t)$ is a near zero variance predictor ($\sigma_\varepsilon \approx 10^{-15}$). Appropriately, ε_t joins to the constant value of target T to model a stochastic target variable.

Now, subject to the equality constraint in (2), consider a class of a MISO system in the following form:

$$\mathbf{x}_t = f(\mathbf{x}_1, \ldots, \mathbf{x}_{t-1}) + g(\mathbf{x}_1, \ldots, \mathbf{x}_{t-1})\mathbf{u}_{t-1} + \mathbf{d}_{t-1}$$
$$\widehat{\mathbf{Q}}_t = \mathbf{x}_t \quad (3)$$

where unknown f and g functions are bounded and no prior knowledge is required for bounding. The state vector of the system assumes an estimate through the optimization process of the control loop. To have a controllable system for the model mentioned in (3), the following assumptions were considered.

Assumption 1: An external disturbance is required to be bound by an unknown constant, which is equivalent to $\lim_{t\to\infty} E(\widehat{\mathbf{Q}}_t) = T + E(\varepsilon_t) = T$.

Assumption 2: It is required $g(\mathbf{x}_1, \ldots, \mathbf{x}_{t-1})$ to be positive, which is equivalent to $\lim_{t\to\infty} Var\ (\widehat{\mathbf{Q}}_t) < \infty$.

In this paper, to optimize the regret function in (2) regarding the state-space model in (3), the LS-SVR is trained to map from the input space to the feature space in the presence of the disturbance $\mathbf{d}_t$. The given kernel function $\kappa(u_t, \mathbf{u})$ handles the mapping model in the feature space. The linear in (4) is called the LSs solution of (2) (for details, see [35])

$$\begin{bmatrix} 0 & \mathbf{I}' \\ \mathbf{I} & \kappa(u,u') + \dfrac{\mathbf{I}}{\mathbb{C}} \end{bmatrix} \begin{bmatrix} 0 \\ (\boldsymbol{\alpha} - \boldsymbol{\alpha}^*) \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{Q} \end{bmatrix}. \quad (4)$$

In the optimal control problem, the aim is to solve the problem (2). To find the optimal control law, one could construct the corresponding approximation function of (3), using the kernel function

$$\widehat{\mathbf{Q}}_t = \sum_{l=1}^{t} \big(-\alpha_l + \alpha_l^*\big)\kappa(u_l, \mathbf{u}) + \widehat{b} \quad (5)$$

where $\widehat{b} = -\sum_{l=1}^{t}(-\alpha_l + \alpha_l^*)\kappa(u_l, \mathbf{u}) + T + \varepsilon_l$, and

$$C_t = (\widehat{\mathbf{Q}}_t - (T+\varepsilon_t))^2 \quad (6)$$

is the cost function which determines how adequately the given noisy model works on the actual data.

B. Random Effect LS-SVR

Now, consider a set of training points $\{(u_{1j} + d_{1j}, T+\varepsilon_1), \ldots, (u_{t-1j} + d_{t-1j}, T+\varepsilon_{t-1})\}$. Based on the theory of best linear unbiased prediction [36], the optimization model in (2) with a random effect for jth factor is formulated as

$$\min\ R_{\text{emp}} = \frac{1}{t}\sum_{l=1}^{t} C_l(T+\varepsilon_l, \widehat{Q}_{lj})$$
$$\text{s.t. } \widehat{Q}_{tj} = \mathbf{h}(\widehat{Q}_{t-1j}, \ldots, \widehat{Q}_{1j}, u_{t-1j}, u_{t-1j} \ldots, z_{t-1j}, \ldots, z_{1j}, d_{tj}). \quad (7)$$

When including the random effects into the LS-SVR model, the single regularization parameter changes into two regularization parameters with one parameter for random error and one parameter for random effects. The random effect parameter vector has $N(0, \Sigma_{z_{lj}j})$ and the error vector $N(0, \Sigma_{C_l j})$ distribution where $\Sigma_{z_{lj}j}$ and $\Sigma_{C_l j}$ are the covariance matrices and are known or could be estimated. Hence, the corresponding dual form of (7) represents the model (8) with random effect (see the Appendix)

$$\begin{bmatrix} 0 & \mathbf{I}' \\ \mathbf{I} & \kappa(u_l, \mathbf{u}) + \dfrac{\mathbf{I}}{\mathbb{C}_1} z'_{l_j} \Sigma^{-1}_{z_{lj}j} z_{lj} + \dfrac{\mathbf{I}}{\mathbb{C}_2} \Sigma^{-1}_{C_l j} \end{bmatrix} \begin{bmatrix} 0 \\ (\alpha - \alpha^*) \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{Q} \end{bmatrix}. \tag{8}$$

The solution for linear model in (8) optimizes the bias $\widehat{b}$ and the support vector $(\alpha_{lj} - \alpha^*_{lj})$. Then, the optimal regression function for the given $\mathbf{u}$ and $\mathbf{z}$ is obtained by

$$\widehat{\mathbf{Q}}_t = \sum_{l=1}^{t} \sum_{j=1}^{N} \big(-\alpha_{lj} + \alpha^*_{lj} \big) \kappa(u_{lj}, \mathbf{u}) + \frac{\mathbf{I}}{\mathbb{C}_1} \sum_{l=1}^{t} \sum_{j=1}^{N} \big(-\alpha_{lj} + \alpha^*_{lj} \big) \Sigma^{-1}_{z_{lj}j} z_{lj} + \widehat{b}. \tag{9}$$

In the system dynamic model (3), the input value of the current run $\mathbf{u}_t$ will be updated by (for more details, see [37])

$$\widehat{u}_{tj} = u_{t-1j} + \frac{E_{t-1j}}{\frac{\partial \kappa(u_{t-1j}, \mathbf{u})}{\partial u_{t-1j}} \sum_{l=1}^{t} \big(-\alpha_{lj} + \alpha^*_{lj} \big) \kappa(u_{lj}, \mathbf{u})}. \tag{10}$$

C. Stability of Random-Effect LS-SVR

The basic idea of stability is that the result of a learning-based system with a full sample should not be very different from the result obtained by removing only one observation. More precisely, for any two subsets of empirical data, the Euclidean distance between the corresponding loss function should be bound by $M \geq 0$.

σ-admissibility [38] condition is considered to impose the stability of a robust algorithm through the learning-based cross-validation scenario.

Definition 3: σ-admissibility condition, a cost function C_t is σ-admissible with respect to the output class $\mathbf{Q}$ if it will be differentiable almost everywhere and there exists $\sigma \in R_+$ such that for any two outputs $Q'_{tj}, Q''_{tj} \in \mathbf{Q}_j$ and all label information $(T + \varepsilon_t)$

$$\big| C_t\big(Q'_{tj}, T + \varepsilon_t\big) - C_t\big(Q''_{tj}, T + \varepsilon_t\big) \big| \leq \sigma \big| Q'_{tj} - Q''_{tj} \big|. \tag{11}$$

This assumption holds for the quadratic cost functions where the set of output and target values is bounded by $M \in R_+ : \forall Q_{tj} \in \mathbf{Q}_j, |Q_{ij}| < M$ and $|T + \varepsilon_t| < M$.

The LS-SVR model with quadratic cost function meets σ-admissible condition with $\sigma = 2$ for any two outputs $Q'_{tj}, Q''_{tj} \in \mathbf{Q}_j$ [38, Corollary 11.1, p. 255].

D. Nonlinear Lyapunov-Based Kernel Function

Inserting a time delay in a loop of the control system causes a reduction in the performance. Moreover, larger values of time delays change the stability of the system. For a stability analysis of time-delay systems (TDSs), the Lyapunov method [39] is known as the most efficient technique. The following Lyapunov theory is validated for all linear TDS systems.

Definition 4: Assume the following Lyapunov function of the system in (3)

$$V_{\text{Lyap}}(t) = \mathbf{x}_t^T \mathbf{P} \mathbf{x}_t \tag{12}$$

the stability of the linear form of dynamic system in (3) can guarantee if and only if

$$\forall \mathbf{A} \in \mathbf{x}_t = \mathbf{A}\mathbf{x}_{t-1} + \mathbf{B}\mathbf{u}_{t-1}$$
$$A_{11}\mathbf{P} + \mathbf{P}A_{11} = -\mathbf{P}^* \tag{13}$$

where $\mathbf{P}$ and $\mathbf{P}^*$ are positive symmetric definite matrices and $(\partial/\partial t) V_{\text{Lap}}(t) < 0$. For the feedforward control system, $\mathbf{P}* = \mathbf{I}$ [40].

For the stability condition of a control system, the characterization of kernel functions and matrices is used in the following way.

Proposition 5 [41]: "Every positive semidefinite and symmetric matrix is a kernel matrix. Conversely, every kernel matrix is symmetric and positive semidefinite."

Therefore, the linear mapping function of Lyapunov condition (12) is invertible, and the stability property for the linear time-invariant system in (3) is under the Lyapunov function where a positive-definite symmetric matrix $\mathbf{P}$ exists, and

$$\big\langle \mathbf{x}, \mathbf{x}' \big\rangle_{\mathbf{p}} = \mathbf{P} \tag{14}$$

can be considered as the inner product of the system in (3) [42].

Generally, in the lithography process, a linear model is modified to illustrate the performance of the exposure tool. Nevertheless, the wafer surface structure abundantly concedes the real distribution of the overlay error as nonlinear and in the curve shape (e.g., due to the upstream process of the exposure step). To deal with this phenomenon, the Lyapunov kernel mapping function of the LS-SVR can achieve the form of the polynomial kernel function

$$\mathbf{P} = \kappa(u_t, \mathbf{u}) = \big(\langle \mathbf{x}, \mathbf{x}' \rangle_{\mathbf{p}} + m \big)^n \tag{15}$$

where $m = 0$, and n are estimated through the tuning procedure. Regarding the dynamic model in (3), an additional condition is required for asymptotic stability of a nonlinear system as follows [39, p. 130]:

$$|f(\mathbf{x}_1, \ldots, \mathbf{x}_{t-1})|^2 \leq \begin{bmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_{t-1} \end{bmatrix} \mathbf{P} \begin{bmatrix} \mathbf{x}_1 & \cdots & \mathbf{x}_{t-1} \end{bmatrix}. \tag{16}$$

IV. Simulation Study

The following steps design the performance of the proposed random effect LS-SVR control system in the presence of metrology delay and high-mixed recipe system in a simulation scenario.

Step 1: Consider $R_1, \ldots, R_5$ as five different recipes and $\mathbf{u}_{ji}$, for $j = 1, \ldots, 10$, $i = 1, \ldots, 5$ as an input variable for jth overlay factor and ith recipe.

Step 2: Generate 1000 samples for each recipe and overlay factors from multivariate Normal distribution as $\mathbf{u}_1, \ldots, \mathbf{u}_5 \sim \text{MVN}(\mu, \Sigma)$ where $\mu_1, \ldots, \mu_5$ are a mean vector of $\vec{\mathbf{0}}, \vec{\mathbf{1}}, \vec{\mathbf{2}}, \vec{\mathbf{3}}, \vec{\mathbf{4}}$ (impulse shift), respectively, with length 10 and Σ is a 10×10 diagonal matrix with diag $\{10, 5, 1, 0.5, 0.1, 0.05, 0.01, 0.005, 0.001, 0.0005\}$ to represent the effect of disturbance on the system.

Step 3: Set stochastic target value almost surely to zero generated by $T + \varepsilon_t \sim N(0, 10^{-15})$.

Step 4: Mix all recipe data randomly to build up a new data set with 5000 instances and ten overlay factors.

Step 5: $z_{tj} = i$ is defined for random effect at tth run such that for jth overlay factor recipe i is used.

Step 6: Simulate stochastic metrology delay from ZIP distribution with ZIP($\lambda = 3, \gamma = 0.9$) where a delay happens with the maximum length of 8 at each lot (25 runs).

Step 7: Generate a sequence of 25 dummy data for $T + \varepsilon_t' \sim T + \varepsilon_t$ for the minimum requirement of the LS-SVR learning algorithm.

Step 8: Maintain five lots of historical data for compensating the overlay error through the learning algorithm.

Step 9: Employ the EWMA and dEWMA controllers as the baseline for the performance comparison. A linear form of EWMA in (17) is assumed for the process output and input estimation at run t

$$\begin{aligned} \widehat{\mathbf{Q}}_t &= a + \tau \mathbf{u}_{t-1} + d_t \\ \mathbf{u}_t &= \mathbf{u}_{t-1} - \frac{\theta}{\hat{\tau}} E_{t-1} \end{aligned} \tag{17}$$

where by the tuning algorithm, the unknown parameters a and τ can be estimated. Based on the engineers' domain knowledge, the discount factor sets to $\theta = 0.3$. Similarly, for dEWMA [19], the process input and output represent as follows:

$$\begin{aligned} \widehat{\mathbf{Q}}_t &= a_t + \tau \mathbf{u}_{t-1} + \mathbf{d}_t \\ \mathbf{u}_t &= \frac{T - a_t - D_t}{\hat{\tau}} \\ a_t &= a_{t-1} + D_{t-1} + \theta_1(\widehat{\mathbf{Q}}_t - \hat{\tau}\mathbf{u}_{t-1} - a_{t-1} - D_{t-1}) \\ D_t &= D_{t-1} + \theta_2(\widehat{\mathbf{Q}}_t - \hat{\tau}\mathbf{u}_{t-1} - a_{t-1} - D_{t-1}). \end{aligned} \tag{18}$$

Step 10: Initiate the tuning algorithm to estimate the best parameter setting of random-effect LS-SVR in (8) with the Lyapunov-based kernel function in (15). The regularization parameters $\mathbb{C}_1$ and $\mathbb{C}_2$ are modified to the interval [0,10], with a lag of 1. m and n in (15) are confirmed to the intervals [-1,1] and [1,10] with a lag of 0.1 and 1, respectively.

Step 11: Adopt the higher first pass rule for R2R-MISO controller in Algorithm 1, in which the input overlay factor with the highest variation from the targeted setpoint enter into the c_1 controller, second highest to the c_2, and so on. The system reaches the steady-state condition if the root-mean-square error (RMSE) condition (19) between two control systems exists

$$|c_j(\text{RMSE}) - c_{j-1}(\text{RMSE})| \leq 10^{-15} \tag{19}$$

where

$$\text{RMSE} = \sqrt{\frac{\sum_{l=1}^{t}[\widehat{\mathbf{Q}}_l - (T + \varepsilon_l)]^2}{t}}.$$

Step 12: Initialize the setting of the control system at a current run for each recipe with the setting of the last run when the same recipe was applied to the system.

Step 13: Merge the tenfold cross-validation learning algorithm to check the optimal setting stability in the LS-SVR controller.

Figs. 5–9 illustrate the comparison of the estimated input and output between the LS-SVR and EWMA and dEWMA controllers. Consequently, the LS-SVR has smoother variations and improved compensation performance (e.g., reduced variance and closer to the target) than both EWMA and dEWMA.

The result shows that the proposed LS-SVR controller tightens up the excellent performance bound, and eventually achieves a lower cost, together with an extensive disturbance, in comparison with the EWMA and dEWMA control system. When the variation increases, the result is more tangible. When the unmeasurable disturbance makes a tangible shift in overlay factors, LS-SVR can competently deal with process shift, while EWMA and dEWMA are inaccurate, although dEWMA has a better performance than EWMA.

On average, after the fifth overlay factor enters into the model, the system reaches the steady-state condition, which means that the proposed LS-SVR controller can compensate the effect of disturbance, noise and impulse shift, smaller than 0.1. However, the result shows that the convergence rate strongly depends on the disturbance or impulse shift.

According to the sensitivity analysis implemented by the tuning procedure, the most frequent result for the parameter setting of the kernel function in (15) achieved at $m = 0$ and $n = 3$. $\mathbb{C}_1$ and $\mathbb{C}_2$ parameters of LS-SVR in (7) are set as the free parameters.

To test how the Lyapunov stability condition and polynomial kernel function effectively enhanced the performance of the LS-SVR method, three types of kernel functions merged with the mixed-effect LS-SVR method (the polynomial kernel function upgraded with the Lyaponuv stability condition, simple polynomial kernel function, and simple linear kernel function). The result is summarized in Table I which shows that both the polynomial kernel function and Lyapunov stability condition are constructive in compensating the overlay factors. It is apparent that when disturbance and impulse shifts are increasing, the performance of the proposed mixed-effect LS-SVR controller is significantly better than the two other LS-SVR controllers.

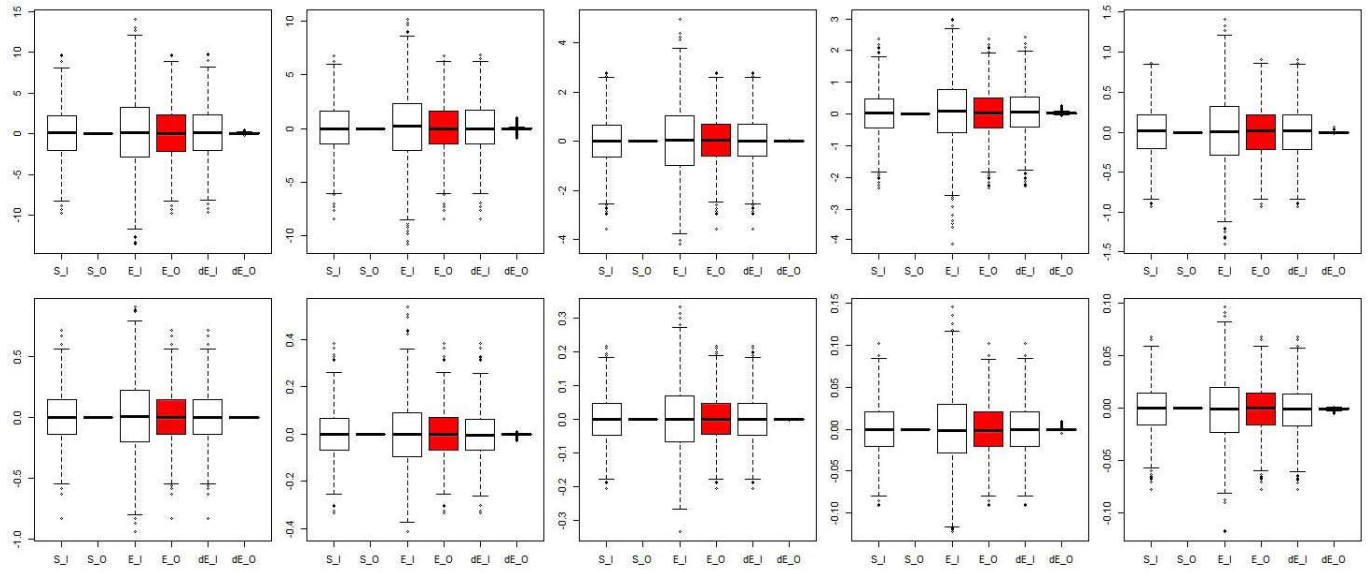


Fig. 5. Simulation result of estimated input and output of ten process variables for random-effect LS-SVR versus EWMA and dEWMA controller from the first recipe (R_1) (S_I denotes to LS-SVR input, S_O to LS-SVR output, E_I to EWMA input, E_O to EWMA output, dE_I to dEWMA input, and dE_O to dEWMA output).

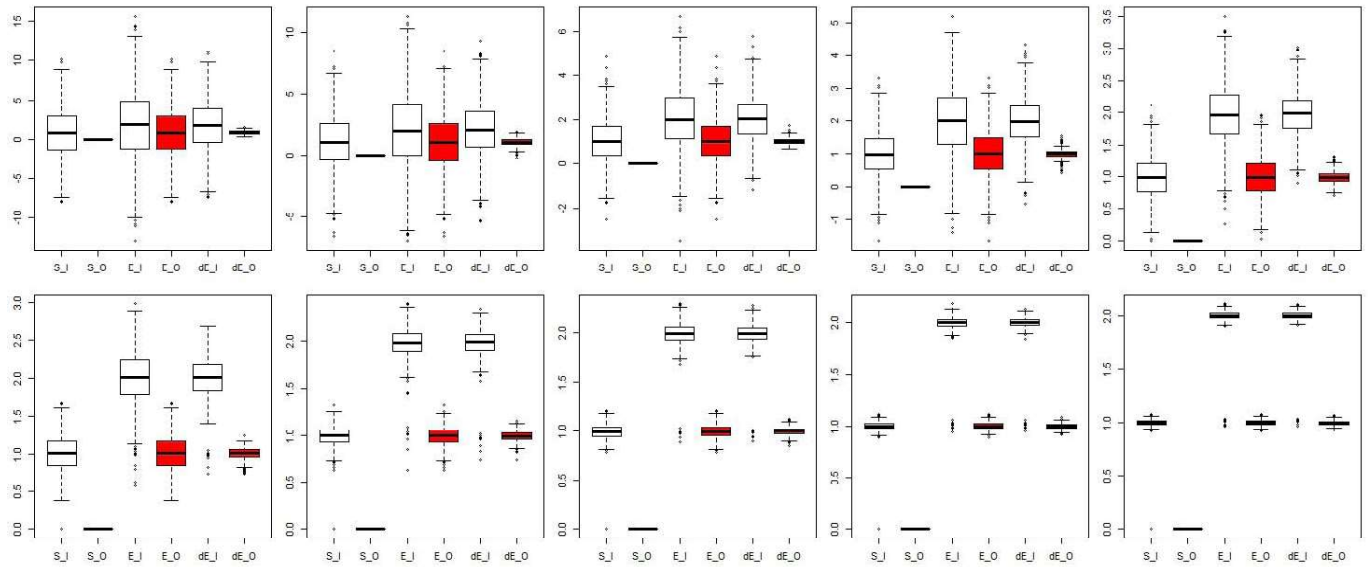


Fig. 6. Simulation results for estimated input and output of ten process variables for random-effect LS-SVR versus EWMA and dEWMA controller from the second recipe (R_2).

V. Empirical Study

In this section, to speed up the analysis of the huge amount of empirical data, the most frequent result from simulation experiment has been used as the initial setting for the optimization model. The general kernel function considers as $\mathbf{P} = (\langle \mathbf{x}, \mathbf{x}' \rangle_{\mathbf{P}})^3$.

The empirical data include four recipes connected to the reticle of the scanner. Among ten overlay factors in [17], asymmetric rotation and asymmetric magnification have not been controlled in this fab. The range (20) and RMSE of empirical data are summarized in Table II. In the measurements, 30% of lots received delay from the metrology tools. The maximum length of the delay is calculated as 80 lots, and the average length is estimated as four lots

$$\text{Range} = \max_t \widehat{Q}_t - \min_t \widehat{Q}_t. \tag{20}$$

The EWMA controller with $\omega = 0.3$ was used for the feedback control in this fab. The target value of each overlay factor and total overlay errors was zeroed. According to the result of the simulation study, variables with variations smaller than 0.1 from the target set point were eliminated from the MISO system. Table III summarizes the improvement of range and RMSE for each overlay factor between EWMA and LS-SVR.

The overlay model proposed in [17] is used to approximate the compounded overlay error on the x-axis and the

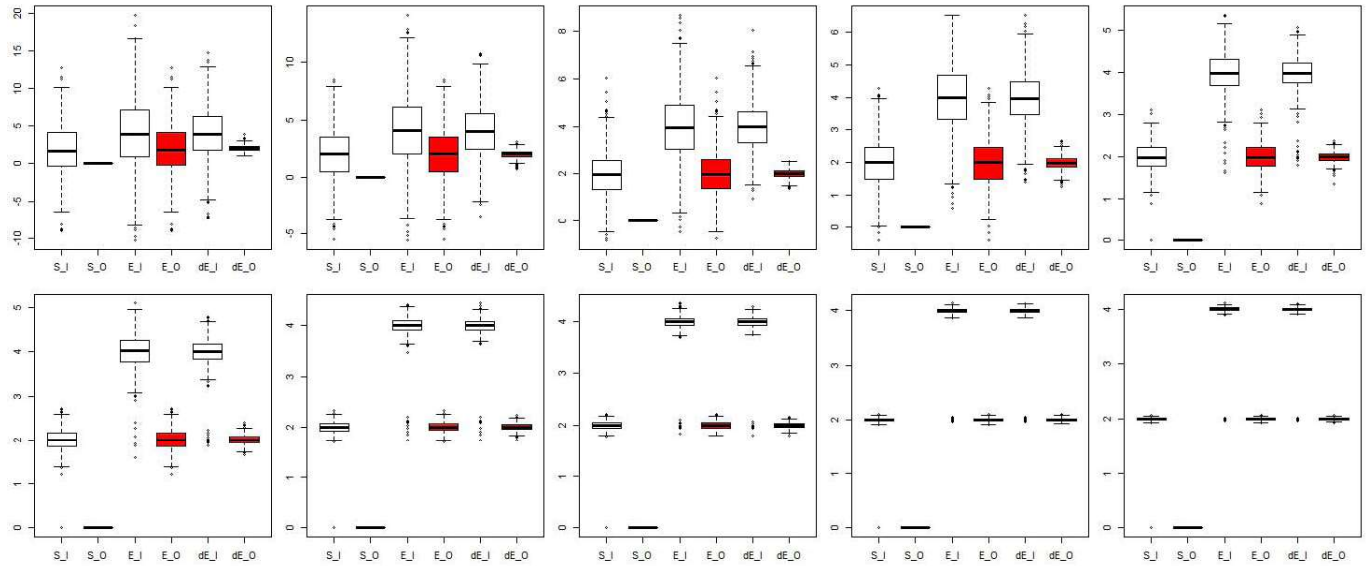


Fig. 7. Simulation result of estimated input and output of ten process variables for random-effect LS-SVR versus EWMA and dEWMA controller from the third recipe (R_3).

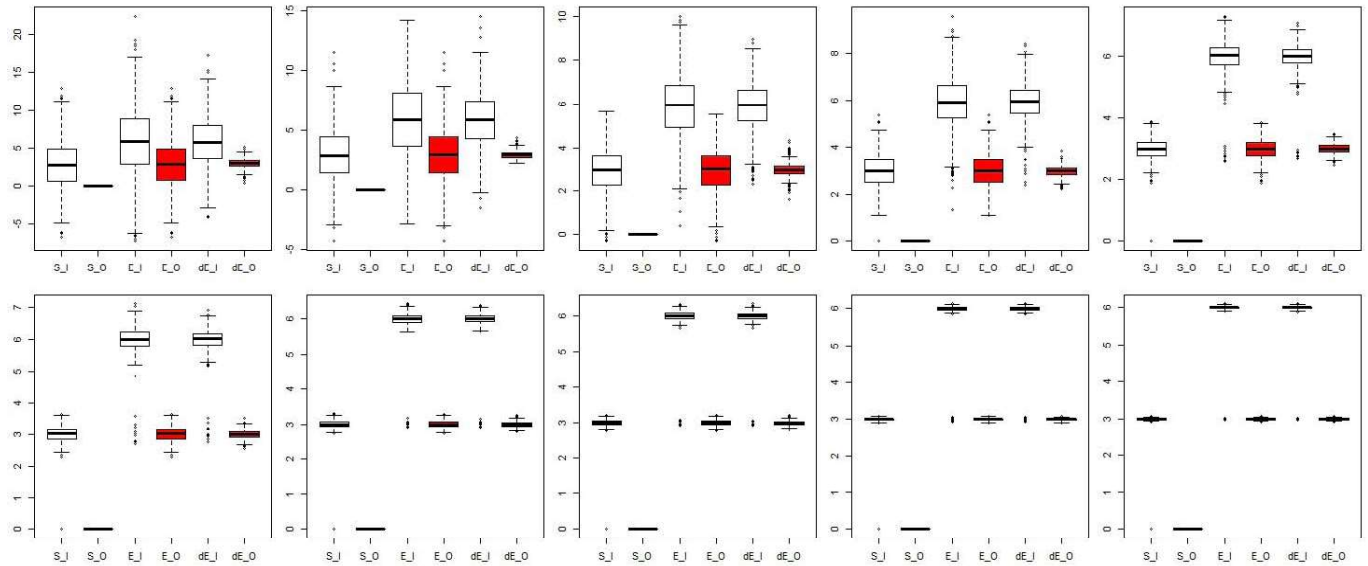


Fig. 8. Simulation result of estimated input and output of ten process variables for random-effect LS-SVR versus EWMA and dEWMA controller from the fourth recipe (R_4).

TABLE I

SIMULATION RESULT OF RANGE AND RMSE IMPROVEMENT COMPARED WITH MIXED-EFFECT LS-SVR WITH LINEAR KERNEL FUNCTION (MIXED-EFFECT LS-SVR WITH POLYNOMIAL KERNEL WITHOUT LYAPUNOV STABILITY CONDITION)

Overlay Factors	Recipe									
	I		II		III		IV		V	
	RMSE	*Range*	*RMSE*	*Range*	*RMSE*	*Range*	*RMSE*	*Range*	*RMSE*	*Range*
V.1	13%(3%)	1%(1%)	14%(4%)	2%(2%)	20%(5%)	5%(4%)	59%(7%)	6%(6%)	61%(9%)	9%(9%)
V.2	4%(2%)	0%(0%)	12%(3%)	0%(0%)	21%(4%)	2%(2%)	23%(7%)	5%(4%)	23%(8%)	6%(6%)
V.3	3%(2%)	0%(0%)	4%(2%)	0%(0%)	7%(3%)	2%(2%)	23%(6%)	4%(3%)	23%(7%)	6%(6%)
V.4	3%(2%)	0%(0%)	4%(2%)	0%(0%)	7%(3%)	2%(1%)	16%(5%)	3%(3%)	17%(6%)	5%(5%)
V.5	3%(2%)	0%(0%)	3%(2%)	0%(0%)	7%(3%)	1%(1%)	13%(4%)	3%(2%)	14%(5%)	5%(4%)
V.6	3%(2%)	0%(0%)	3%(2%)	0%(0%)	7%(3%)	1%(1%)	9%(3%)	2%(2%)	14%(5%)	5%(2%)
V.7	3%(2%)	0%(0%)	3%(2%)	0%(0%)	7%(3%)	0%(0%)	9%(3%)	2%(1%)	10%(4%)	2%(2%)
V.8	3%(2%)	0%(0%)	3%(2%)	0%(0%)	7%(3%)	0%(0%)	9%(3%)	1%(0%)	9%(3%)	1%(1%)
V.9	3%(2%)	0%(0%)	3%(2%)	0%(0%)	4%(2%)	0%(0%)	7%(2%)	0%(0%)	7%(2%)	1%(1%)
V.10	1%(1%)	0%(0%)	1%(1%)	0%(0%)	3%(1%)	0%(0%)	6%(1%)	0%(0%)	6%(1%)	1%(1%)

y-axis. At each run, only the maximum value of measurement error for the x-axis and the y-axis is considered. The result is summarized in Table IV and shows how the error compensation has improved by using the proposed mixed-effect LS-SVR controller for each recipe. Our results show that the proposed LS-SVR controller has achieved an improvement of a minimum 32% on the indices.

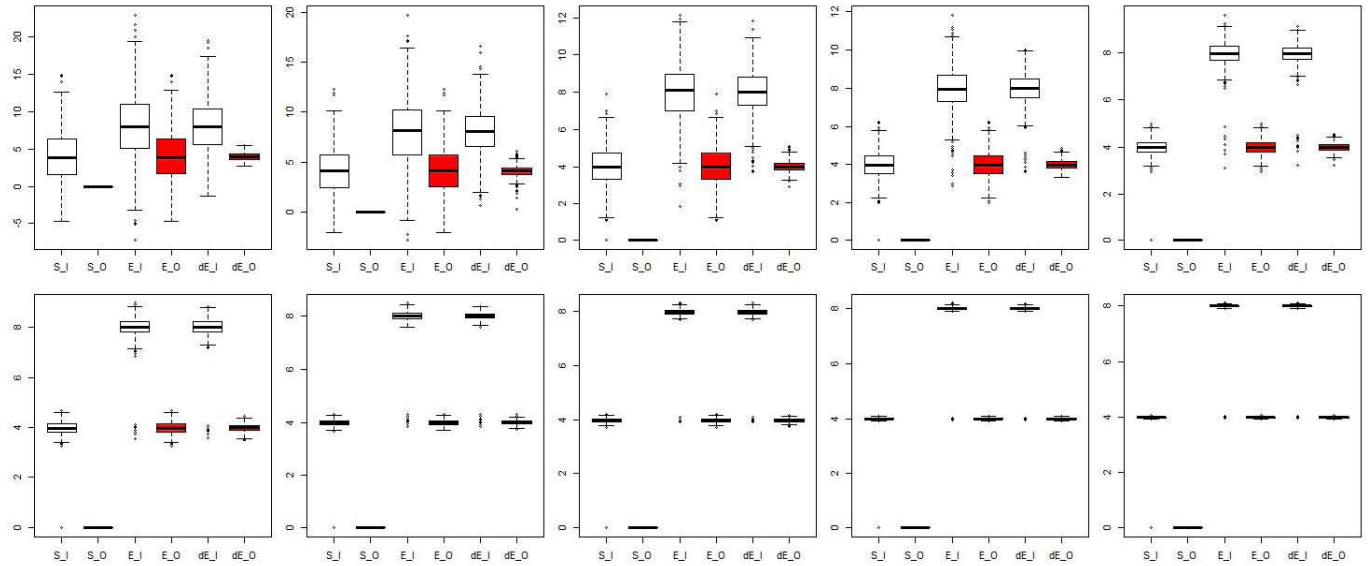


Fig. 9. Simulation result of estimated input and output of ten process variables for random-effect LS-SVR versus EWMA and dEWMA controller from the fifth recipe (R_5).

TABLE II
SUMMARY OF EMPIRICAL DATA

Overlay Factors	Recipe (number of lot)							
	A (154)		B (25)		C (3)		D (2)	
	RMSE	*Range*	*RMSE*	*Range*	*RMSE*	*Range*	*RMSE*	*Range*
Wafer rotation	7.02	68.71	7.32	59	6.47	37.1	6.8	29.54
Non-orthogonality	2.79	33.42	2.49	16.61	1.89	9	2.17	11.89
Scaling along the X axis	6.71	52.9	6.91	68.45	5.79	25.2	7.45	31.47
Scaling along the Y axis	7.61	62.5	7.45	62.22	6.97	40.2	7.79	32.03
Translation along the X axis	7.89	54.6	7.7	41.5	6.78	30.6	8.72	48.2
Translation along the Y axis	8.37	74	8	48.8	6.94	41.3	7.15	30.1
Reticle rotation	3.23	30.04	3	19.7	2.47	13.99	3.32	16.48
Isotropic magnification	2.98	37.57	2.66	19.03	2.41	12.34	4.5	27.68

TABLE III
RANGE AND RMSE IMPROVEMENT

Overlay Factors	Recipe (number of lot)							
	A (154)		B (25)		C (3)		D (2)	
	RMSE	*Range*	*RMSE*	*Range*	*RMSE*	*Range*	*RMSE*	*Range*
Wafer rotation	92%	84%	100%	90%	83%	139%	157%	171%
Non-orthogonality	51%	90%	64%	59%	40%	74%	41%	104%
Scaling along the X axis	54%	53%	52%	71%	52%	69%	43%	40%
Scaling along the Y axis	54%	65%	56%	61%	70%	123%	37%	43%
Translation along the X axis	57%	49%	39%	46%	50%	37%	35%	60%
Translation along the Y axis	57%	60%	46%	49%	29%	44%	23%	26%
Reticle rotation	43%	60%	102%	77%	23%	38%	34%	37%
Isotropic magnification	29%	49%	25%	45%	59%	35%	52%	93%

TABLE IV
IMPROVEMENT FOR OVERLAY ERROR ON x-AXIS AND y-AXIS

Overlay Factors	Recipe (number of lot)							
	A (154)		B (25)		C (3)		D (2)	
	RMSE	*Range*	*RMSE*	*Range*	*RMSE*	*Range*	*RMSE*	*Range*
X-axis	11%	8%	-8%	0%	15%	14%	14%	26%
Y-axis	17%	23%	16%	21%	18%	11%	18%	30%

VI. CONCLUSION

This paper developed an accurate, high efficient optimization technique for overlay error minimization during the high-mixed photolithography process as the basis for productivity improvements in digital communications and industrial transformation [43]. The proposed mixed-effect LS-SVR controller with a self-tuning algorithm combined with polynomial Lyapunov-based kernel function shows its robust capability to compensate the overlay error. The assumed Lyapunov condition composes a stable controller to deal with the lack of process information caused by the metrology delay.

It takes a long time for tuning the algorithm's input parameters (m, n, $\mathbb{C}_1$, and $\mathbb{C}_2$) to converge to the best parameter setting. Specifically, there are only two sensitive parameters ($\mathbb{C}_1$, and $\mathbb{C}_2$, the regularization parameters) which should be estimated. For achieving a more accurate result, the three mixed-effect LS-SVR controllers with the polynomial Lyapunov-based kernel, a simple polynomial kernel, and simple linear kernel functions have been compared.

The result of the simulation study shows that both the polynomial kernel and the Lyapunov stability function enhance the performance of the proposed control system.

In addition, the empirically collected data from a leading semiconductor fab in Taiwan have been used to validate the proposed LS-SVR controller.

In comparison to the EWMA or dEWMA, the LS-SVR method is enforced the historical data for the algorithm learning process enhancement, such that more training information will be affected positively by the performance of the control system. In this paper, the information of at least last five lots was used training as data. The proposed approach could be extended to multilayer overlay error processes by considering an additional random effect in each layer. The mixed-recipe process is directly applicable to the mixed-product process, and with a few adjustments is relevant to the mixed-tool process. The high-mixed recipe-product process only requires an additional random effect.

Moreover, not only this method is applicable to overlay errors modeling in the semiconductor industry but also other processes in dry-etching (DE), chemical-mechanical polishing (CMP), and similar industries such as panel, textile, and solar cell industry.

Future Research Topics

In summary, some further remarks and future research topics are listed as follows.

1) Extending the proposed model to other semiconductor manufacturing process such as DE or CMP.
2) Developing the model for the multilayer lithography process.
3) Extending the high-mixed model for any mixture-system (i.e., mixtures of recipes, products, and tools).

APPENDIX

Regards to the model with random-effects, the optimization problem for LS-SVR can be equivalently written as

$$\min_{w,C_l,b,\boldsymbol{\beta}_l} \frac{1}{2}\|w\|^2 + \frac{\mathbb{C}_1}{2}\sum_{l=1}^{t}\beta_{\mathbf{l}}'\Sigma_{z_{lj}j}^{-1}\beta_{\mathbf{l}} + \frac{\mathbb{C}_2}{2}\sum_{l=1}^{t}C_{lj}\,\Sigma_{C_lj}^{-1}C_{lj'}$$
$$\text{s.t. } w\phi\,(\boldsymbol{u}_l) + \boldsymbol{\beta}_l\boldsymbol{z}_l + +C_l + b = T + \varepsilon_{lj}. \quad (21)$$

This is a convex quadratic program with affine constraints. (22) is introducing the corresponding Lagrangian function

$$L(w, C_l, b, \boldsymbol{\beta}_l; \boldsymbol{\alpha}, \boldsymbol{\alpha}^*) = \frac{1}{2}\|w\|^2 + \frac{\mathbb{C}_1}{2}\sum_{l=1}^{t}\beta_{\mathbf{l}}'\Sigma_{z_{lj}j}^{-1}\beta_{\mathbf{l}} + \frac{\mathbb{C}_2}{2}\sum_{l=1}^{t}C_{lj}\,\Sigma_{C_lj}^{-1}C_{lj'} - \sum_{l=1}^{t}\boldsymbol{\alpha}_l(T + \boldsymbol{\varepsilon}_l - w\phi(\boldsymbol{u}_l) - \boldsymbol{\beta}_l\boldsymbol{z}_l - b - C_l) - \sum_{l=1}^{t}\boldsymbol{\alpha}_l^*(T + \boldsymbol{\varepsilon}_l - w\phi(\boldsymbol{u}_l) - \boldsymbol{\beta}_l\boldsymbol{z}_l - b - C_l). \quad (22)$$

The KKT optimization conditions for a solution can be achieved by partially differentiating concerning w, C_l, b, $\boldsymbol{\beta}_l$, $\boldsymbol{\alpha}$, and $\boldsymbol{\alpha}^*$

$$\begin{cases} \dfrac{\partial L}{\partial w} = 0 \rightarrow \quad w = \displaystyle\sum_{l=1}^{t}(\alpha_l - \alpha_l^*)\phi(\mathbf{u}_l) \\ \dfrac{\partial L}{\partial b} = 0 \rightarrow \quad \displaystyle\sum_{l=1}^{t}(\alpha_l - \alpha_l^*) = 0 \\ \dfrac{\partial L}{\partial C_l} = 0 \rightarrow \quad \mathbb{C}_2\Sigma_{C_lj}^{-1}C_l - (\boldsymbol{\alpha}_l - \boldsymbol{\alpha}_l^*) = 0 \\ \dfrac{\partial L}{\partial \alpha_l} = 0 \rightarrow \quad \displaystyle\sum_{l=1}^{t}(T + \varepsilon_l - w\phi(\mathbf{u}_l) - \boldsymbol{\beta}_l z_l - b - C_l) = 0 \\ \dfrac{\partial L}{\partial \alpha_l^*} = 0 \rightarrow \quad \displaystyle\sum_{l=1}^{t}(T + \varepsilon_l - w\phi(\mathbf{u}_l) - \boldsymbol{\beta}_l z_l - b - C_l) = 0 \\ \dfrac{\partial L}{\partial \boldsymbol{\beta}_l} = 0 \rightarrow \quad \mathbb{C}_1\Sigma_{z_{lj}j}^{-1}\beta_l - \alpha_l z_l = 0 \end{cases} \quad (23)$$

which leads to the following equivalent dual problem regarding the kernel matrix $\kappa(\mathbf{u}_l, \mathbf{u}) = \phi(\mathbf{u}_l)\phi^T(\mathbf{u}_l)$

$$\begin{bmatrix} 0 & \mathbf{I}' \\ \mathbf{I} & \kappa(u_l, \mathbf{u}) + \dfrac{\mathbf{I}}{\mathbb{C}_1}z_{l_j}'\,\Sigma_{z_{lj}j}^{-1}z_{lj} + \dfrac{\mathbf{I}}{\mathbb{C}_2}\Sigma_{C_lj}^{-1} \end{bmatrix}\begin{bmatrix} 0 \\ (\alpha - \alpha^*) \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{Q} \end{bmatrix}. \quad (24)$$

REFERENCES

[1] M. Khakifirooz, C. F. Chien, and Y.-J. Chen, "Bayesian inference for mining semiconductor manufacturing big data for yield enhancement and smart production to empower industry 4.0," *Appl. Soft Comput.*, vol. 68, pp. 990–999, Jul. 2018.

[2] Y. Zheng, D. S.-H. Wong, Y.-W. Wang, and H. Fang, "Takagi–Sugeno model based analysis of EWMA RtR control of batch processes with stochastic metrology delay and mixed products," *IEEE Trans. Cybern.*, vol. 44, no. 7, pp. 1155–1168, Jul. 2014.

[3] A.-C. Lee, J.-H. Horng, T.-W. Kuo, and N.-H. Chou, "Robustness analysis of mixed product run-to-run control for semiconductor process based on ODOB control structure," *IEEE Trans. Semicond. Manuf.*, vol. 27, no. 2, pp. 212–222, May 2014.

[4] C. E. Chemali, J. Freudenberg, M. Hankinson, and J. J. Bendik, "Run-to-run critical dimension and sidewall angle lithography control using the PROLITH simulator," *IEEE Trans. Semicond. Manuf.*, vol. 17, no. 3, pp. 388–401, Aug. 2004.

[5] H.-H. Ko, J.-S. Kim, J. Kim, J.-G. Baek, and S.-S. Kim, "Intelligent adaptive process control using dynamic deadband for semiconductor manufacturing," *Expert Syst. Appl.*, vol. 38, no. 6, pp. 6759–6767, 2011.

[6] Y. Jiao and D. Djurdjanovic, "Stochastic control of multilayer overlay in lithography processes," *IEEE Trans. Semicond. Manuf.*, vol. 24, no. 3, pp. 404–417, Aug. 2011.

[7] H.-F. Kuo and A. Faricha, "Artificial neural network for diffraction based overlay measurement," *IEEE Access*, vol. 4, pp. 7479–7486, 2016.

[8] M. Khakifirooz, M. Fathi, and C.-F. Chien, "Modelling and decision support system for intelligent manufacturing: An empirical study for feedforward-feedback learning-based run-to-run controller for semiconductor dry-etching process," *Int. J. Ind. Eng. Theory, Appl. Practice*, vol. 25, no. 6, 2018.

[9] H. Narita, "Semiconductor device manufacturing method and semiconductor device," U.S. Patent 9 269 671, Feb. 23, 2016.

[10] K. Nara and T. Hamada, "Method for manufacturing display element, manufacturing apparatus of display element and display device," U.S. Patent 9 310 656, Apr. 12, 2016.
[11] F. He and Z. Zhang, "An empirical study-based state space model for multilayer overlay errors in the step-scan lithography process," *RSC Adv.*, vol. 5, no. 126, pp. 103901–103906, 2015.
[12] S. C. Horng, "Compensating modeling overlay errors using the weighted least-squares estimation," *IEEE Trans. Semicond. Manuf.*, vol. 27, no. 1, pp. 60–70, Feb. 2014.
[13] H. Fuyun and Z. Zhang, "State space model and numerical simulation of overlay error for multilayer overlay lithography processes," in *Proc. 2nd Int. Conf. Image, Vis. Comput. (ICIVC)*, Jun. 2017, pp. 1123–1127.
[14] D. MacMillen and W. D. Ryden, "Analysis of image field placement deviations of a 5×microlithographics reduction lens," *Proc. SPIE*, vol. 0334, pp. 78–89, Sep. 1982.
[15] C.-F. Chien, K.-H. Chang, and C.-P. Chen, "Design of a sampling strategy for measuring and compensating for overlay errors in semiconductor manufacturing," *Int. J. Prod. Res.*, vol. 41, no. 11, pp. 2547–2561, 2003.
[16] C.-F. Chien and C.-Y. Hsu, "A novel method for determining machine subgroups and backups with an empirical study for semiconductor manufacturing," *J. Intell. Manuf.*, vol. 17, no. 4, pp. 429–439, 2006.
[17] C.-F. Chien and C.-Y. Hsu, "UNISON analysis to model and reduce step-and-scan overlay errors for semiconductor manufacturing," *J. Intell. Manuf.*, vol. 22, no. 3, pp. 399–412, 2011.
[18] J. Park *et al.*, "Exact and reliable overlay metrology in nanoscale semiconductor devices using an image processing method," *J. Micro/Nanolithography, MEMS, MOEMS*, vol. 13, no. 4, pp. 041409-1–041409-7, 2014.
[19] A.-C. Lee, T.-W. Kuo, and C.-T. Ma, "Combined product and tool disturbance estimator for the mix-product process and its application to the removal rate estimation in CMP process," *Int. J. Precis. Eng. Manuf.*, vol. 13, no. 4, pp. 471–481, 2012.
[20] A.-C. Lee, T.-W. Kuo, and S.-W. Chiang, "Run-to-run mixed product overlay process control: Using tool based disturbance estimator (TBDE) approach," *Int. J. Eng. Technol.*, vol. 5, no. 3, p. 349, 2013.
[21] B. Ai, Y. Zheng, Y. Wang, S.-S. Jang, and T. Song, "Cycle forecasting ewma (CF-EWMA) approach for drift and fault in mixed-product run-to-run process," *J. Process Control*, vol. 20, no. 5, pp. 689–708, 2010.
[22] R. P. Good and S. J. Qin, "On the stability of MIMO EWMA run-to-run controllers with metrology delay," *IEEE Trans. Semicond. Manuf.*, vol. 19, no. 1, pp. 78–86, Feb. 2006.
[23] B. J. de Kruif and T. J. A. de Vries, "On using a support vector machine in learning feed-forward control," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics*, vol. 1, Jul. 2001, pp. 272–277.
[24] J. A. Suykens, J. Vandewalle, and B. De Moor, "Optimal control by least squares support vector machines," *Neural Netw.*, vol. 14, no. 1, pp. 23–35, 2001.
[25] S. K. Lahiri and K. C. Ghanta, "Support vector regression with parameter tuning assisted by differential evolution technique: Study on pressure drop of slurry flow in pipeline," *Korean J. Chem. Eng.*, vol. 26, no. 5, pp. 1175–1185, 2009.
[26] C. L. Smith, "Split-range control," *Adv. Process Control*, pp. 86–125, 2010.
[27] J. Rico-Azagra, M. Gil-Martínez, and J. Elso, "Quantitative feedback control of multiple input single output systems," *Math. Problems Eng.*, vol. 2014, Apr. 2014, Art. no. 136497.
[28] T. Liu and F. Gao, "Cascade control system," in *Industrial Process Identification and Control Design*. Springer, 2012, pp. 321–347.
[29] B. Lu *et al.*, "Improving process monitoring and modeling of batch-type plasma etching tools," Ph.D. dissertation, Dept. Chem. Eng., Univ. Texas, Austin, TX, USA, 2015.
[30] Y. E. Shao and Y.-T. Hu, "Using the ANN classifier to recognize the disturbance patterns for a multivariate system," in *Proc. 5th IIAI Int. Congr. Adv. Appl. Informat. (IIAI-AAI)*, Jul. 2016, pp. 630–634.
[31] X. Cheng, X. Ren, S. Ma, and S. Li, "A modeling method based on artificial neural network with monotonicity knowledge as constraints," *Chemometrics Intell. Lab. Syst.*, vol. 145, pp. 93–102, Jul. 2015.
[32] T. Sun and J. Liu, "Predicting MEMS gyroscope's random drifts using LSSVM optimized by modified PSO," in *Proc. IEEE Chin. Guid., Navigat. Control Conf.*, Aug. 2016, pp. 146–149.
[33] B. Ahmadi, H. Nourisola, and S. Tavakoli, "Robust adaptive sliding mode control design for uncertain stochastic systems with time-varying delay," *Int. J. Dyn. Control*, vol. 6, no. 1, pp. 413–424, 2018.
[34] S. R. Mohanty, P. K. Ray, N. Kishor, and B. Panigrahi, "Classification of disturbances in hybrid DG system using modular PNN and SVM," *Int. J. Elect. Power Energy Syst.*, vol. 44, no. 1, pp. 764–777, 2013.
[35] V. Vapnik, *Statistical Learning Theory*. New York, NY, USA: Wiley, 1998.
[36] G. K. Robinson, "That BLUP is a good thing: The estimation of random effects," *Statist. Sci.*, vol. 6, no. 1, pp. 15–32, 1991.
[37] X. Yuan, Y. Wang, and L. Wu, "Composite feedforward-feedback controller for generator excitation system," *Nonlinear Dyn.*, vol. 54, no. 4, pp. 355–364, 2008.
[38] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*. Cambridge, MA, USA: MIT Press, 2012.
[39] E. Fridman, *Introduction to Time-Delay Systems: Analysis and Control*. Springer, 2014.
[40] W. Zhang, M. S. Branicky, and S. M. Phillips, "Stability of networked control systems," *IEEE Control Syst. Mag.*, vol. 21, no. 1, pp. 84–99, Feb. 2001.
[41] G. R. G. Lanckriet, N. Cristianini, P. Bartlett, L. El Ghaoui, and M. I. Jordan, "Learning the kernel matrix with semidefinite programming," *J. Mach. Learn. Res.*, vol. 5, pp. 27–72, Jan. 2004.
[42] I. Kalashnikova, B. van Bloemen Waanders, S. Arunajatesan, and M. Barone, "Stabilization of projection-based reduced order models for linear time-invariant systems via optimization-based eigenvalue reassignment," *Comput. Methods Appl. Mech. Eng.*, vol. 272, pp. 251–270, Apr. 2014.
[43] M. Khakifirooz, D. Cayard, C. F. Chien, and M. Fathi, "A system dynamic model for implementation of industry 4.0," in *Proc. Int. Conf. Syst. Sci. Eng. (ICSSE)*, Jun. 2018, pp. 1–6.

Marzieh Khakifirooz (S'–) received the M.S. degree in industrial statistics and the Ph.D. degree in industrial engineering from National Tsing Hua University (NTHU), Hsinchu, Taiwan, in 2014 and 2018, respectively.

She is currently a Post-Doctoral Researcher with the Artificial Intelligence for Intelligent Manufacturing Systems Research Center, sponsored by the Ministry of Science and Technology, NTHU. Her research interests include smart manufacturing, game theory, big data analytics, statistical inferences, and operational research.

Chen-Fu Chien (M'–) received the B.S. degree (Phi Tao Phi Hons.) with double majors in industrial engineering and electrical engineering from National Tsing Hua University (NTHU), Hsinchu, Taiwan, in 1990, and the M.S. degree in industrial engineering and the Ph.D. degree in operations research and decision sciences from the University of Wisconsin–Madison, Madison, WI, USA, in 1994 and 1996, respectively.

From 2002 to 2003, he was a Fulbright Scholar with the University of California at Berkeley, Berkeley, CA, USA. From 2005 to 2008, he was on-leave as the Deputy Director with the Industrial Engineering Division, Taiwan Semiconductor Manufacturing Company (TSMC), Taiwan. He received the PCMPCL Training from the Harvard Business School, Boston, MA, USA, in 2007. He is currently a Tsinghua Chair Professor and a Micron Chair Professor with NTHU. He is also the Director of the Artificial Intelligence for Intelligent Manufacturing Systems Research Center sponsored by the Ministry of Science and Technology, NTHU-TSMC Center for Manufacturing Excellence, and the Principal Investigator for the NSC Semiconductor Technologies Empowerment Partners Consortium. He holds eight invention patents on semiconductor manufacturing. He has published four books, more than 170 journal papers, and a number of case studies in Harvard Business School. His research efforts centre on decision analysis, modeling and analysis for semiconductor manufacturing, manufacturing strategy, and manufacturing intelligence.

Dr. Chien was a recipient of the National Quality Award, the Executive Yuan Award for Outstanding Science and Technology Contribution, the Distinguished Research Awards, the Tier 1 Principal Investigator (Top 3%) from the Ministry of Science and Technology, the Distinguished University-Industry Collaborative Research Award from the Ministry of Education, the University Industrial Contribution Awards from the Ministry of Economic Affairs, the Distinguished University-Industry Collaborative Research Award, the Distinguished Young Faculty Research Award from NTHU, the Distinguished Young Industrial Engineer Award, the Best IE Paper Award, the IE Award from Chinese Institute of Industrial Engineering, the Best Engineering Paper Award, the Distinguished Engineering Professor by Chinese Institute of Engineers in Taiwan, the 2012 Best Paper Award of the IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING, and the 2015 Best Paper Award of the IEEE TRANSACTIONS ON SEMICONDUCTOR MANUFACTURING. He is an Area Editor of the *Flexible Services and Manufacturing Journal*, an Editorial Board Member of *Computers and Industrial Engineering*, and an Advisory Board Member of *OR Spectrum*. He has been invited to give keynote lectures at several conferences, including APIEMS, C&IE, FAIM, IEEM, IEOM, IML, and leading universities worldwide.

Mahdi Fathi (M'–) received the B.S. and M.S. degrees from the Department of Industrial Engineering and Management Systems, Amirkabir University of Technology, Tehran, Iran, in 2006 and 2008, respectively, and the Ph.D. degree from the Iran University of Science and Technology, Tehran, in 2013.

Dr. Fathi received five post-doctoral fellowships at Industrial Engineering Laboratory, Ecole Central Paris, Châtenay-Malabry, France, in 2014; Stochastic Modeling and Analysis of Communication Systems Group at the Department of Telecommunications and Information Processing, Ghent University, Ghent, Belgium, in 2015; Department of Engineering Systems and Design, Singapore University of Technology and Design, Singapore, in 2016; Center for Applied Optimization, University of Florida, Gainesville, FL, USA, in 2017, Department of Industrial and Systems Engineering, Mississippi State University, Starkville, MS, USA, in 2018. In summary, his vision is "To Disrupt Industrial and System Engineering Field with New Type of AI Solutions, Optimization and Big Data Analytics Moving Toward New Era of Smart Manufacturing Post Technology 4.0." His research interests are queuing theory and its applications, stochastic process, optimization, artificial intelligent, smart manufacturing, and reliability. AQ:15

A fuzzy clustering-based method for scenario analysis in strategic planning: The case of an Asian pharmaceutical company

M.S. Pishvaee and M. Fathi
Department of Industrial Engineering, Amirkabir University of Technology,
Tehran, Iran
ms-pishvaee@aut.ac.ir and mfathi_ieyahoo.com

F. Jolai*
Department of Industrial Engineering, Faculty of Engineering,
University of Tehran, Iran
fjolai@ut.ac.ir

Received August 2008

In today's rapid changing market situations, many nations and companies try to keep or make better their situation and gain more market share by creating competitive advantages. Because of growing number of uncertain parameters in the environment and lack of information about the future, the strategic choice has become very complex and critical. One of the popular tools for solving the problem is scenario analysis. In this paper based on fuzzy clustering we propose a method for building, analyzing and ranking the possible scenarios. To cope with the issue of uncertain parameters of the environment in strategic planning, we use the concept of fuzzy set theory to enhance the proposed method. Finally the performance of the proposed method is illustrated in a strategic planning case in a pharmaceutical company.

*To whom all correspondence should be addressed.

Introduction

In today's world, the market climate changes more quickly and countries realize that globalization makes the world smaller and more competitive. Also customers seek products and services that can respond to their specific needs and firms attempt to create competitive advantages to keep their profit and market share. Complex business organizations and competitive environment uncertainty have focused business researchers' and practitioners' attention on the use of strategic planning to integrate and optimize management processes. These trends compel firms and countries to forecast future events and design a proper action for future trends.

Strategic planning helps an organization to cope with increasing environmental turbulence and complexity, more intense competitive pressures, and the pace of technological change (McDonald, 1992). The main focus of strategic planning is deciding what strategies should be used to create a sustainable competitive advantage for competing in a given product market.

So for making effective and sustained decisions in strategic planning in an uncertain environment, knowing the current condition of environment and forecasting the future trends of environmental factors is necessary and critical. The environment can be regarded as a number of different compartments and processes that interact in a complex system. The assessment of the environmental consequences of an event is therefore complicated by the variety of influencing factors (Andersson, Stjernstrom & Fangmark, 2005). But because of rapid environmental changes, intensified competitiveness and uncertainty of environmental factors, making or formulating strategies based on traditional methods is become very difficult and in some cases impossible.

To cope with this issue in strategic planning, a popular tool is scenario analysis, whereby scenarios are built for possible events in future and strategies planned with respect to these scenarios.

A scenario is a description of a future situation and the course of events that enables one to progress from the original situation to the future situation. The word scenario is often abused, especially when used to describe any set of hypotheses. Of course, these hypotheses must simultaneously be pertinent, coherent, plausible, important, and transparent to meet all of desirable criteria (Godet, 2000). Qualitative scenarios describe possible futures in the form of words or symbols, while quantitative scenarios describe futures in numerical form (Alcamo, 2001). A good scenario should be relevant, consistent (coherent), probable and transparent. In principle, only a few substantially different scenarios are needed (Nguyen, De Kok & Titus, 2006). The participatory approach to scenario building, which is widely acknowledged, requires a wide spectrum of knowledge and opinions from multidisciplinary team

members (Schwab, Cerutti & Von Reibnitz, 2003; Van der Heijden, 1996).

Scenario analysis has three main purposes. The first is to forecast the environment; the second is to evaluate strategic options, especially the robustness of them, against the possible scenarios; and the third is to provide a non-technical audience a picture of future alternative states of the environment in an easily understandable form that can provide an effective format in which information in both qualitative and quantitative forms can be assimilated and represented (Derek & Ahti, 1993; Nguyen *et al*., 2006).

But one problem is that scenarios build on experts' opinions and usually experts forecast future in linguistic expressions. So the knowledge and the experience of experts are often the data sources in strategic planning (Sarin, 1979). Another problem is the complexity associated with uncertain environmental factors and often lack of relevant historical data. The complexity of the environmental problems makes necessary the development and application of new tools capable of processing not only the numerical aspects, but also the experience of experts and wide public participation, which are all needed in the decision-making process. This paper aimed to develop a simple methodology to cope with this problem. The main contribution of this paper lies in the implementation of a simple methodology based on fuzzy set theory and fuzzy clustering aimed to assist decision makers in their strategic decision process.

The reason for using fuzzy set theory and fuzzy clustering stems from the complexity, uncertainty and lack of knowledge associated with environment. The fuzzy set theory is precisely a theory that provides a framework to handle the sources of uncertainty, including vagueness, ambiguity and imprecision, at the same time (Nguene & Finger, 2007). Fuzzy set theory that was originally developed by Zadeh (1973), aims to formalize the linguistic reasoning in mathematical form, which provides a means of approximate characterization of phenomena that are too complex to be amenable to description in conventional quantitative terms.

Many papers have used fuzzy set theory in management issues like transportation (e.g. Sheu, 2005), logistics or supply chain (e.g. Wang & Shu, 2007; Hu & Sheu, 2003) and strategic management (e.g. Kardaras & Karakostas, 1999), but few papers have applied fuzzy set theory to scenario analysis (Nguyen *et al*., 2006; Nguene & Finger, 2007).

Using a fuzzy clustering approach in aggregating the possible scenarios into a few main scenarios and developing a method for calculating the consistency or compatibility of scenarios is what differentiates this research with the past works mentioned above.

The remainder of this paper is organized as follows. In Section 2 we describe the proposed methodology. For better understanding of the proposed method, a case study is presented in Section 3, and in Section 4 a summary of the work and some possible future works are presented.

Methodology

To cope with the issues of linguistic expression of an expert in strategic planning, uncertainties and lack of information about future forecasting, we use the concept of fuzzy set theory and fuzzy clustering to develop a hybrid method for scenario analysis. This method comprises five main steps:

Step 1 Defining key factors and describing their possible future trends in the opinion of experts.

Step 2 Generating all possible scenarios from a combination of factors' future trends.

Step 3 Calculating pair wise compatibility indexes and eliminating incompatible scenarios.

Step 4 Defining main scenarios with Fuzzy C-means clustering (FCM) method and ranking them by calculating the degree of possibility for each final scenario.

Step 5 Project the cluster centres for expressing and interpreting the main scenarios in linguistic terms.

The input of this system is qualitative expressions by several experts about key factors' future trends (step 1). Then these subjective qualitative expressions are translate into a quantitative form based on fuzzy numbers for generating all possible scenarios (steps 1 and 2). Scenario analysis and compatibility measurements are performed by fuzzy set theory operators (step 3) and by aggregating the possible compatible scenarios based on fuzzy clustering the main final scenarios are achieved. Finally, the main scenarios are presented and used in both quantitative and qualitative (linguistic expression) form. This process is illustrated in Figure 1.

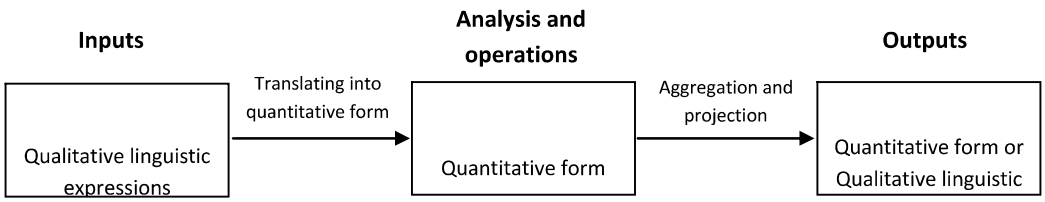


Figure 1: Methodology process

The detail of each of the five steps is described in below.

Defining the key factors and their future trends

In this step, based on environmental analysis and experts' opinions, the most influential key factors (variables) are distilled. In strategic planning literature these factors often classified in five segments: Political, Economic, Cultural and Social, Technological, Demographic (Pearce & Robinson, 2005; Mowen & Minor, 1997). Consequently, we should define the main factors with respect to these areas and forecast their future trends by experts' opinions. But as mentioned earlier, one problem is that experts forecast the future in linguistic expressions (e.g. market demand for lifestyle drugs will be increased strongly). To cope with this issue we use fuzzy set theory. If the market demand is a linguistic variable, the value terms of this linguistic variable could be "low", "moderate" and "high" instead of crisp numbers. A linguistic expression like "market demand for lifestyle drugs will be high" could be modeled by a fuzzy number. A fuzzy number is a normal and convex fuzzy set with bounded support (Klir & Yuan, 2002). The fuzzification, which can be described by the process of establishment of membership functions, requires several steps, consisting of the establishment of ranges in the numerical domains of the key factors concerned, the specification of boundaries in the fuzzy domains of associated fuzzy subsets and the selection of the shape of the membership functions (MFs) (Nguyen *et al*., 2006)*. There are, in general, no rules for the selection of shape of a membership function when little data and expert's knowledge about a variable exist. Therefore, the symmetrically trapezoidal, triangular and Gaussian MFs are often chosen for this purpose. In this paper MFs of key factors have the triangular form (Figure 2).

In addition to the specification of the numerical ranges of variables, it is necessary to specify the boundaries of the associated fuzzy subsets. For example, from what value to what value can the market demand be considered to be ''low'' or ''high''.

The boundaries of fuzzy subsets may interred, i.e. one particular demand quantity can belong to both ''low'' and ''medium'' fuzzy subsets (Figure 2). These boundaries are often established subjectively from the experience of experts.

Generating all possible scenarios

Now it is possible to generate the scenarios. Each combination of forecasted future trends of key factors makes a possible scenario. As illustrated in the case presented in Figure 3, the combination of forecasted trends makes 834 possible scenarios. So the number of scenarios grows very fast when the number of forecasted trends increases.

Many of these possible scenarios have a very low degree of possibility; we named them in this paper "incompatible scenarios". For a valid strategic planning, incompatible scenarios should be detected and eliminated. For this purpose, a method is proposed based on fuzzy set theory in the next section.

Calculating the fuzzy compatibility index and eliminating incompatible scenarios

One way to measure the degree of possibility of a scenario is to measure the compatibility of each pair of forecasted future trends in each scenario. However the relation between the pair wise forecasted trends often can not be clearly expressed by an expert, especially for long-term forecasting. Moreover there are no past data to refer to. Therefore, to determine the degree of possibility of a scenario that shows its importance, we define a fuzzy compatibility index (FCI) between each pair of forecasted future trends. The concept of triangular fuzzy number (TFN) is used for this purpose. As shown in Figure 4, five linguistic variables as TFNs between $\tilde{1}$ and $\tilde{5}$ ($FCI_{(i,\ j)}$) were defined. FCIs were determined based on experts' opinions.

Experts determine the compatibility between each pair of forecasted trends in linguistic terms in a range of "very low" to "very high" (very low, low, medium, high and very high). For eliminating the incompatible scenarios two rules were used:

1. If one or more than one of FCIs between each pair of forecasted trends ($FCI_{(i,\ j)}$) in a scenario is "very low" the related scenario was eliminated.

2. If the average of FCIs in a scenario is less than "Medium" (or fuzzy number $\tilde{3}$), the related scenario was eliminated.

At the end of this stage we have some compatible scenarios, but as mentioned before if there are too many scenarios (more than 5 or 6), they will lose their characteristics and blur the main issue. Therefore to solve this problem and to achieve the efficient number of scenarios, similar scenarios should be grouped. We use fuzzy clustering for this purpose as described in the next section.

Defining the final scenarios and their ranking

Classification methods can be divided in two main groups of techniques: discriminant analysis and cluster analysis. Cluster analysis refers to the unsupervised situation where little or no information is available about group structure prior to the classification. The goal is to find groups in the data. Discriminant analysis refers to situations where the membership of a set of samples is known and the main purpose is to build a classification rule applicable for new and unknown samples.

*For more information about the concepts of the MFs we refer readers to (Zadeh, 1973).

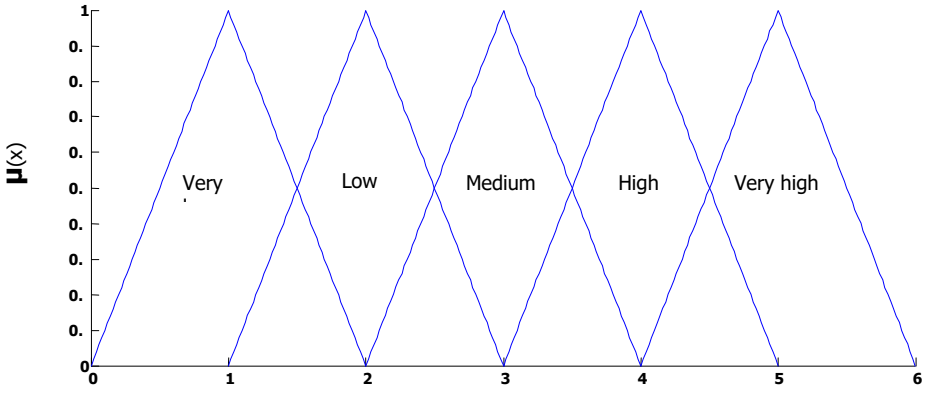


Figure 2: Graphical representation of alinguistic variable

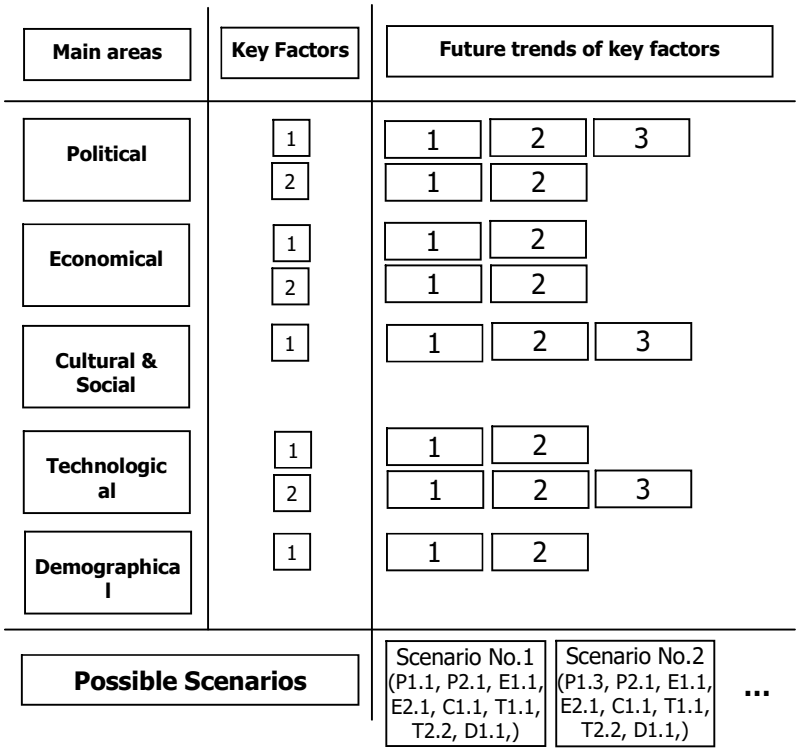


Figure 3: Generating possible scenarios

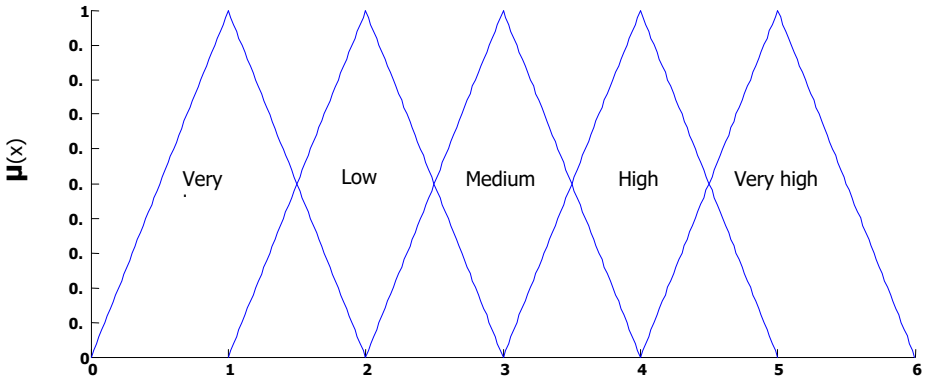


Figure 4: Linguistic variables for FCI

Cluster analysis is based on partitioning a collection of data points into a number of subgroups, where the objects inside a cluster (a subgroup) show a certain degree of closeness or similarity. Hard clustering assigns each data point to one and only one of the clusters, with a degree of membership equal to one, assuming well defined boundaries between the clusters. This model does not often reflect the description of real data, where boundaries between subgroups might be fuzzy and where a more nuanced description of object affinity to the specific cluster is required (Gath & Geva, 1989).

Fuzzy clustering, as an advanced clustering method, unlike the hard clustering methods, allows each data point to belong to several groups with different degrees of similarity bounded within the range of 0 and 1. Another benefit is that in fuzzy clustering we can analyze multi-dimensional data consisting of linguistic attributes (Hu & Sheu, 2003).

The Fuzzy C-means clustering algorithm (FCM) is based on the minimization of an objective function called C-means functional (Bezdek, 1975). The FCM algorithm uses a standard Euclidean distance norm, which induces hyper spherical clusters.

For the Fuzzy C-Means clustering algorithm there are three input parameters needed to run this function:

- The number of clusters or initializing partition matrix.
- The fuzziness weighting exponent.
- The maximum termination tolerance.

The two latter parameters have their default value, if they are not given by the user. The function calculates with the standard Euclidean distance norm, the norm inducing matrix at an (n x n) identity matrix. The result of the partition is collected in structure arrays.

In this paper we use FCM algorithm in our methodology to group the similar scenarios. The detail of the FCM algorithm is described in the appendix.

Another issue is determining the number of clusters that represent the number of final scenarios. It is important that we know how many clusters should be used in FCM method. For this purpose, a cluster validity method must be examined.

Validity methods began with Bezdek partition coefficient (V_{PC}) and partition entropy (V_{PE}) of U matrix, which shows the degree of membership of each data to each cluster. In this paper we use a more recent index known as the Fukuyama-Sugeno index. This index provided better response versus other validity indexes include partition coefficient and partition entropy of Bezdek (Pal & Bezdek, 1994). The index is presented in the appendix. To choose the optimal number of clusters that shows the number of final scenarios, we proposed the below procedure.

Number of clusters NC=1

Maximum allowed number of scenarios MN=max (MN is a case based number usually between 5 to 7)

$FSI_1=0$

While $FSI_{(NC)}<FSI_{(NC-1)}$ and NC<MN

NC=NC+1

Calling FCM algorithm procedure

Calculating Fukuyama-Sugeno index $FSI_{(NC)}$

After clustering the scenarios we have the main scenarios, but these scenarios do not have the same degree of possibility, because of the difference in their degree of possibility. To rank these main scenarios with respect to their importance (degree of possibility) we propose a formula as follow:

$$RS(h) = \frac{\sum_{k=1}^{n} \sum_{i=1}^{m} \sum_{j=1}^{m} \mu_k \; FCI_{(i,j)}}{n} \qquad \text{... (1)}$$

n: number of compatible scenarios in each cluster.

m: number of forecasted future trends.

μ_k : Degree of membership of scenario k to cluster h.

This formula could properly represent the degree of possibility of the main scenarios. The scenarios that achieve a higher ranking score (RS) are more important than others, because the possibility of occurrence for these scenarios is higher than the others.

Expressing and interpreting the main scenarios

Ideally we should present the scenarios in linguistic expressions to senior management for decision making. To express the final scenarios in a linguistic statement we introduce a method base on the projection of cluster centres. In this method cluster centres were projected on each axis. Each axis is related to one key factor and as mentioned before we define linguistic variables on each axis base on fuzzy numbers. Therefore each cluster centre as a main scenario represents a scenario that includes all the key factors and their future trends. As a result we could explain each main scenario in linguistic expression with this method. In the next section the proposed method is used in a case study.

Case study

In this section the proposed method is applied to a case study in the pharmaceutical market. The studied case is a pharmaceutical Asian company with two plants and one research centre. The company is produces three drugs. Two of them are usual prescription drugs with an approximately smooth demand used in medical care activities. These drugs

are under protection of government and direct government pricing rules. There is a possibility of exporting for these two drugs to two neighboring countries, which have a critical need for these drugs. But this opportunity is encountering some political obstacles. The other drug (X) is an over-the-counter lifestyle drug without government protection and with free market pricing. Drug X is developed in the company's research centre and the company has patent rights to it. X sales is almost 45% of the company's total revenues. One critical problem that threatens future X sales is the weakness of government laws in patent protection. This problem seems more important because now company decides to develop another lifestyle drug in its research center. The other critical factor is the size and structure of market demand. There are several forecasted future trends for this key factor because of environmental uncertainty. This has a significant effect on the company's strategic choice. If the share of lifestyle drugs in the structure of demand increases or decreases the strategic choice will change significantly. We define the MFs of each key factor base on expert opinions. For example the MF of "share of lifestyle drugs in total demand" is illustrated in Figure 5. Other key factors and their forecasted future trends are illustrated in Table 1.

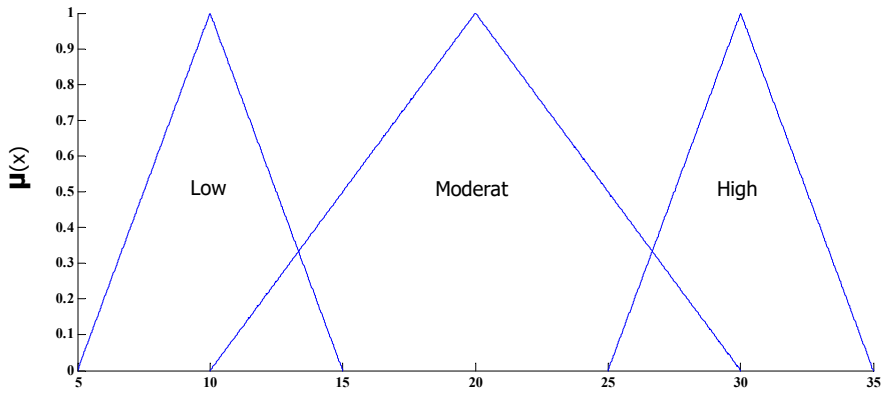


Figure 5: Share of lifestyle drugs in total demand (%) MF

Table 1: Key factors in Pharmaceutical industry and their future trends determined by experts

Main Areas	Key factors	Future trends				
		VL	L	M	H	VH
Political-Legal	P1. Possibility of export to neighbor countries		●		●	
	P2. Extension of health care insurance by Gov.			●		●
Economical	E1. Rate of merging in Pharmaceutical industry		●		●	
	E2. Share of lifestyle drugs in total demand		●	●	●	
	E3. Total demand			●		●
Cultural-Social	C1. Culture of consuming lifestyle drugs		●		●	
Demographical	D1. Population Senility				●	

Table 2: All possible scenarios

Scenario No.	Forecasted trends													
	P1.1	P1.2	P2.1	P2.2	E1.1	E1.2	E2.1	E2.2	E2.3	E3.1	E3.2	C1.1	C1.2	D1.1
1	P1.1		P2.1		E1.1		E2.1			E3.1		C1.1		D1.1
2	P1.2		P2.1		E1.1		E2.1			E3.1		C1.1		D1.1
.	.		.		.		.			.		.		.
.	.		.		.		.			.		.		.
.	.		.		.		.			.		.		.
46	P1.2		P2.1		E1.1		E2.3			E3.2		C1.2		D1.1
47	P1.1		P2.1		E1.2		E2.3			E3.2		C1.2		D1.1
48	P1.2		P2.1		E1.2		E2.3			E3.2		C1.2		D1.1
.	.		.		.		.			.		.		.
.	.		.		.		.			.		.		.
.	.		.		.		.			.		.		.
95	P1.1		P2.2		E1.2		E2.3			E3.2		C1.2		D1.1
96	P1.2		P2.2		E1.2		E2.3			E3.2		C1.2		D1.1

Table 3: The pairwise FCIs between forecasted future trends

	P1.1	P1.2	P2.1	P2.2	E1.1	E1.2	E2.1	E2.2	E2.3	E3.1	E3.2	C1.1	C1.2	D1.1
P1.1			$\tilde{3}$	$\tilde{3}$	$\tilde{3}$	$\tilde{3}$	$\tilde{2}$	$\tilde{3}$	$\tilde{4}$	$\tilde{3}$	$\tilde{1}$	$\tilde{3}$	$\tilde{3}$	$\tilde{3}$
P1.2			$\tilde{3}$	$\tilde{3}$	$\tilde{3}$	$\tilde{4}$	$\tilde{4}$	$\tilde{3}$	$\tilde{2}$	$\tilde{1}$	$\tilde{5}$	$\tilde{3}$	$\tilde{3}$	$\tilde{3}$
P2.1					$\tilde{3}$	$\tilde{3}$	$\tilde{3}$	$\tilde{3}$	$\tilde{3}$	$\tilde{4}$	$\tilde{3}$	$\tilde{3}$	$\tilde{3}$	$\tilde{3}$
P2.2					$\tilde{3}$	$\tilde{4}$	$\tilde{4}$	$\tilde{3}$	$\tilde{2}$	$\tilde{3}$	$\tilde{4}$	$\tilde{3}$	$\tilde{3}$	$\tilde{4}$
E1.1							$\tilde{3}$	$\tilde{3}$	$\tilde{2}$	$\tilde{3}$	$\tilde{3}$	$\tilde{3}$	$\tilde{2}$	$\tilde{3}$
E1.2							$\tilde{3}$	$\tilde{3}$	$\tilde{4}$	$\tilde{3}$	$\tilde{4}$	$\tilde{3}$	$\tilde{4}$	$\tilde{3}$
E2.1										$\tilde{2}$	$\tilde{4}$	$\tilde{4}$	$\tilde{1}$	$\tilde{2}$
E2.2										$\tilde{3}$	$\tilde{3}$	$\tilde{3}$	$\tilde{3}$	$\tilde{3}$
E2.3										$\tilde{4}$	$\tilde{2}$	$\tilde{2}$	$\tilde{4}$	$\tilde{4}$
E3.1												$\tilde{3}$	$\tilde{3}$	$\tilde{3}$
E3.2												$\tilde{2}$	$\tilde{4}$	$\tilde{4}$
C1.1														$\tilde{2}$
C1.2														$\tilde{4}$
D1.1														

As explained before, now all possible scenarios must be generated from the combination of forecasted future trends. In this case with respect to Table 1, the combination of forecasted trends makes 96 possible scenarios (see Table 2). To omit the incompatible scenarios, first FCIs between forecasted trends should be determined based on experts' opinions. Pairwise FICs for the pharmaceutical company case are illustrated in Table 3. After determining FCIs, we can now eliminate the incompatible scenarios using the two rules proposed in Section 2.3.

After exerting the rules on possible scenarios of the studied case, 35 scenarios remained (see Table 4). But as mentioned before, for a good scenario analysis the number of scenarios should not exceed from 6, otherwise they will lose their characteristics and blur the main issue. Therefore, as described in Section 2.4 we use the FCM fuzzy clustering method to group the similar scenarios. The results of clustering are shown in Table 5. The optimum number of clusters that represent the number of final main scenarios in this case is 5. As mentioned before this number is calculated on the basis of the Fukuyama-Sugeno validity index. Now we can project the cluster centres to present the final main scenarios in linguistic terms. Final main scenarios and their related ranking scores are shown in Table 6. Ranking scores show the degree of possibility of each scenario.

Table 4: Eliminating the incompatible scenarios

Scenario No.	Scenario Value	Elimination Rule 1	Elimination Rule 2	Scenario No.	Scenario Value	Elimination Rule 1	Elimination Rule 2
1	2,904	not rejected	rejected	**49**	2,952	not rejected	rejected
2	2,904	rejected	rejected	**50**	2,952	rejected	rejected
3	2,952	not rejected	rejected	**51**	3	not rejected	not rejected
4	3,142	rejected	not rejected	**52**	3,047	rejected	not rejected
5	2,85	rejected	rejected	**53**	3	rejected	not rejected
6	3,14	not rejected	not rejected	**54**	3,285	not rejected	not rejected
7	2,904	rejected	rejected	**55**	3,095	rejected	not rejected
8	3,238	not rejected	not rejected	**56**	3,428	not rejected	not rejected
9	2,809	rejected	rejected	**57**	2,857	rejected	rejected
10	2,809	rejected	rejected	**58**	2,857	rejected	rejected
11	2,904	rejected	rejected	**59**	3	rejected	not rejected
12	2,952	rejected	rejected	**60**	3,047	rejected	not rejected
13	2,857	rejected	rejected	**61**	3	rejected	not rejected
14	3,142	rejected	not rejected	**62**	3,285	rejected	not rejected
15	3	rejected	not rejected	**63**	3,190	rejected	not rejected
16	3,333	rejected	not rejected	**64**	3,523	rejected	not rejected
17	3	not rejected	not rejected	**65**	3	not rejected	not rejected
18	2,904	rejected	rejected	**66**	2,904	rejected	rejected
19	3	not rejected	not rejected	**67**	3,047	not rejected	not rejected
20	2,952	rejected	rejected	**68**	3	rejected	not rejected
21	2,857	rejected	rejected	**69**	2,952	rejected	rejected
22	3,047	not rejected	not rejected	**70**	3,142	not rejected	not rejected
23	2,904	rejected	rejected	**71**	3,047	rejected	not rejected
24	3,142	not rejected	not rejected	**72**	3,285	not rejected	not rejected
25	3,047	not rejected	not rejected	**73**	3,047	not rejected	not rejected
26	2,952	rejected	rejected	**74**	2,952	rejected	rejected
27	3,142	not rejected	not rejected	**75**	3,190	not rejected	not rejected
28	3,095	rejected	not rejected	**76**	3,142	rejected	not rejected
29	3	rejected	not rejected	**77**	3,095	rejected	not rejected
30	3,190	not rejected	not rejected	**78**	3,285	not rejected	not rejected
31	3,142	rejected	not rejected	**79**	3,285	rejected	not rejected
32	3,523	not rejected	not rejected	**80**	3,523	not rejected	not rejected
33	3,047	not rejected	not rejected	**81**	3	not rejected	not rejected
34	2,857	rejected	rejected	**82**	2,809	rejected	rejected
35	3,095	not rejected	not rejected	**83**	3,142	not rejected	not rejected
36	2,9045	rejected	rejected	**84**	3	rejected	not rejected
37	2,809	rejected	rejected	**85**	2,857	rejected	rejected
38	2,904	not rejected	rejected	**86**	2,952	not rejected	rejected
39	2,952	rejected	rejected	**87**	3,047	rejected	not rejected
40	3,095	not rejected	not rejected	**88**	3,190	not rejected	not rejected
41	3,190	not rejected	not rejected	**89**	3,142	not rejected	not rejected
42	3	rejected	not rejected	**90**	2,952	rejected	rejected
43	3,380	not rejected	not rejected	**91**	3,380	not rejected	not rejected
44	3,238	rejected	not rejected	**92**	3,238	rejected	not rejected
45	3,047	rejected	not rejected	**93**	3,095	rejected	not rejected
46	3,142	not rejected	not rejected	**94**	3,190	not rejected	not rejected
47	3,285	rejected	not rejected	**95**	3,380	rejected	not rejected
48	3,428	not rejected	not rejected	**96**	3,523	not rejected	not rejected

Table 5: Cluster centres

Cluster centres	Forecasted trends						
	P1	P2	E1	E2	E3	C1	D1
No.1	3.2109	94.9015	29.7549	29.7424	11.7109	3.4991	40
No.2	3.0092	94.8687	29.6747	14.1130	11.5092	2.4952	40
No.3	2.5632	60.0398	29.7551	22.4906	11.4632	2.8239	40
No.4	2.5603	94.9044	10.0680	22.3129	11.4603	3.0363	40
No.5	2.5629	60.0536	10.2252	22.5126	11.4629	3.0328	40

Table 6: The final main scenarios and their ranking score

Main Final Scenarios	Forecasted trends							RS
	P1	P2	E1	E2	E3	C1	D1	
No.1	Medium	High	High	High	High	High	High	21.0132
No.2	Medium	High	High	Low	Medium	Low	High	19.3064
No.3	Low	Medium	High	Medium	Medium	Medium	High	24.5509
No.4	Low	High	Low	Medium	Medium	Medium	High	22.7351
No.5	Low	Medium	Low	Medium	Medium	Medium	High	24.0394

Results of clustering show that in this case, five main scenarios could explain the possible future situation in the Pharmaceutical market. As is shown in Table 6 scenario No.3 is the most possible scenario. We could explain main scenario No.3 in a linguistic statement as follow.

There is a low possibility of exporting to two neighboring countries and the health care insurance will be extended moderately. Pharmaceutical companies will merge, especially with regard to research laboratories. Population senility will grow highly. Total demand, share of lifestyle drugs and the culture of consuming lifestyle drugs will grow moderately.

Conclusion

Scenario analysis is a powerful tool to cope with environment changes in strategic planning. But it associates with two problems, one is that scenarios build on expert's opinions and almost experts forecast future in linguistic expressions and second is the complexity associated with uncertain environmental factors and often lack of relevant historical data.

In this paper we introduce a method for scenario analysis in strategic planning to cope with the issues of uncertain parameters of environment and linguistic expression of an expert in strategic planning. We use fuzzy set theory and fuzzy clustering method to represent and group the expert's linguistic expressions. The proposed method is used for scenario analysis in a case in Pharmaceutical market to illustrate the performance. At the end we propose the following issues for future researches.

- Using the fuzzy clustering method that works with fuzzy data.
- Using fuzzy multi criteria decision making (FMCDM) methods for scenario ranking.
- Relation of scenario analysis and strategy ranking.
- Relation of scenario analysis and the robustness of strategies.

References

Alcamo, J. 2001. *Scenarios as tools for international environmental assessments*. Environmental Issue Report, No. 24. Experts' Corner Report. Prospects and Scenarios No. 5. Copenhagen, Denmark: European Environment Agency.

Andersson, A.S., Stjernstrom, O. & Fangmark, I. 2005. 'Use of questionnaires and an expert panel to judge the environmental consequences of chemical spills for the development of an environment-accident index', *Journal of Environmental Management,* **75**: 247–261.

Bezdek, J.C. & Dunn, J.C. 1975. 'Optimal fuzzy partitions: A heuristic for estimating the parameters in a mixture of normal distributions', *IEEE Trans. Comp*. **C-24**:835-838.

Bezdek, J.C. 1981. *Pattern recognition with fuzzy objective function algorithms*. New York: Plenum Press.

Derek, W.B. & Athi, A.S. 1993. 'Forecasting with scenarios', *European Journal of Operational Research*, **68**:291-303.

Hu, T.L. & Sheu, J.B. 2003. 'A fuzzy-based customer classification method for demand-responsive logistical distribution operations', *Fuzzy Sets and Systems*, **139**(2):431-450.

Kardaras, D. & Karakostas, B. 1999.'The use of fuzzy cognitive maps to simulate the information systems strategic planning process', *Information and Software Technology,* **41**:197–210

Klir, G.J. & Yuan, B. 2002. *Fuzzy sets and fuzzy logic theory and application*. India: Prentice Hall of India.

Gath, I. & Geva, A.B. 1989. 'Unsupervised optimal fuzzy clustering', *IEEE Trans. Pattern Anal. Machine Intell*. PAM 1-**11**(7):773-781.

Godet, M. 2000. 'The art of scenarios and strategic planning: Tools and pitfalls', *Technological Forecasting and Social Change*, **65**:3–22.

McDonald, M.H.B. 1992. 'Strategic marketing planning: A state-of-the-art review', *Marketing Intelligence and Planning*, **10**:4–22.

McDonald, M. 1996. 'Strategic marketing planning: Theory, practice and research agendas', *Journal of Marketing Management*, **12**: 5–27.

Mowen J.C. & Minor, M.J. 1997. *Consumer behavior*. New York, Prentice-Hall.

Nguene, G.N. & Finger, M. 2007. 'A fuzzy-based approach for strategic choices in electric energy supply: The case of a Swiss power provider on the eve of electricity market opening', *Engineering Applications of Artificial Intelligence,* **20**:37–48.

Nguyen, T.G., De Kok, J.L. & Titus, M.J. 2006. 'A new approach to testing an integrated water systems model using qualitative scenarios', *Environmental Modelling & Software,* **22**(11):1557-1571.

Pal, N.R. & Bezdek, J.C. 1994. 'On cluster validity for Fuzzy c-Means Model'. Submitted to *IEEE Transactions on Fuzzy Systems*.

Pearce, L.A. & Robinson R.B. 2005. *Strategic management: Formulation, implementation and control*. NewYork, McGraw-Hill.

Sarin, R.K. 1979. 'An approach to cross impact analysis', *Futures*, **10**:543-554.

Schwab, P., Cerutti, F. & Von Reibnitz, C. 2003. 'Foresight-using scenarios to shape the future of agriculture research', *Foresight* **5**(1):55e61.

Sheu, J.B. 2005. 'A fuzzy clustering approach to real-time demand-responsive bus dispatching', *Fuzzy Sets and Systems*, **15**: 437-455.

Subhash, J.C. 1990. *Marketing planning and strategy.* 3rd Edition. Cincinnati, OH: South-Western Publishing Co.

Van der Heijden, K. 1996. *Scenarios: The art of strategic conversation*. Chichester: John Wiley & Sons.

Wang, J. & Shu Y. 2007. 'A possibilistic decision model for new product supply chain design', *European Journal of Operational Research* **177**: 1044-1061.

Wang, H.F. 1999. 'A fuzzy approach to scenario analysis in strategic planning'. **In** *IEEE International Fuzzy Systems Conference Proceedings, Seoul, Korea.*

Zadeh L.A. 1973. 'Outline of a new approach to the analysis of complex systems and decision processes', *IEEE Transactions on Systems*, Man, Cybernetics, SMC-3, 28e34.

Appendix

FCM Clustering Algorithm

Given the data set X which includes X and Y, the number of clusters 1<c<N, the weighting exponent m>1, the termination tolerance ε>0 and the norm-inducing matrix A, the algorithm tracks the following steps.

Step 1: Calculate the cluster centers

$$V_i^{(l)} = \frac{\sum_{k=1}^{N} (\mu_{ik}^{(l-1)})^m x_k}{\sum_{k=1}^{N} (\mu_{i,k}^{(l-1)})^m}, 1 \le i \le c \quad (1)$$

Step 2: Compute the distances:

$$D_{ikA}^2 = (x_k - v_i)^T A(x_k - v_s), 1 \le i \le c, 1 \le k \le N. \quad (2)$$

Step 3: Update the partition matrix:

$$\mu_{i,k}^{(l)} = \frac{1}{\sum_{j=1}^{c} (D_{ikA} / D_{jkA})^{2/(m-1)}} \quad (3)$$

This steps will be repeated for l =1,2,… until

$$\| U^{(l)} - U^{(l-1)} \| < \varepsilon$$

Fukuyama-Sugeno validity index

Let U∈Mfc and $v = (v_1, v_2, ..., v_c)$ be vector of distinct points vi ∈ R^p for1≤ i ≤ C (here they are cluster centers). Fukuyama and Sugeno presents a new cluster validity index ($\mathbf{V_{FS}}$) as follows:

$$V_{FS}(U,V;X) = \sum_{i=1:C} \sum_{k=1:N} \left(U_{i,k}\right)^m \left(\|x_k - v_i\|^2 - \|v_i - \bar{v}\|^2\right) \quad (1)$$

A good (U,V) pair should produce a small value of the index.

Journal of Manufacturing Systems 29 (2010) 55–62

Contents lists available at ScienceDirect

Journal of Manufacturing Systems

journal homepage: www.elsevier.com/locate/jmansys

Technical paper

A queueing approach to production-inventory planning for supply chain with uncertain demands: Case study of PAKSHOO Chemicals Company

E. Teimoury [a,b], M. Modarres [c], F. Ghasemzadeh [d,e], M. Fathi [a,*]

[a] Department of Industrial Engineering, Iran University of Science and Technology, Tehran, Iran
[b] Logistics & Supply Chain Researches & Studies Group, Institute for Trade Studies & Research, Tehran, Iran
[c] Department of Industrial Engineering, Sharif University of Technology, Tehran, Iran
[d] Department of Industrial Engineering, Industrial Management Institute, Tehran, Iran
[e] Production Planning Expert, PAKSHOO Chemicals Company, Tehran, Iran

ARTICLE INFO

Article history:
Received 18 May 2010
Received in revised form
16 August 2010
Accepted 16 August 2010
Available online 17 September 2010

ABSTRACT

In some industries such as the consumable product industry because of small differences between products made by various companies, customer loyalty is directly related to the availability of products required at that time. In other words, in such industries demand cannot be backlogged but can be totally or partly lost. So companies of this group use make-to-stock (MTS) production policy. Therefore, in these supply chains, final product warehouses play a very important role, which will be highlighted by considering the demand uncertainty as it happens in real world, especially in the consumable product industries in which demand easily varies according to the customer's taste variation, behavioral habits, environmental changes, etc. In this article, an (s, Q) inventory system with lost sales and two types of customers, ordinary and precedence customers and exponentially distributed lead times are analyzed. Each group of demands arrives according to the two independent Poisson processes with different rates. A computationally efficient algorithm for determining the optimal values for safety stock as reorder level and reorder quantity for a multi-item capacitated warehouse is developed. The algorithm also suggests the optimal warehouse capacity. A Multi-item Capacitated Lot-sizing problem with Safety stock and Setup times (MCLSS) production planning model is then developed to determine the optimal production quantities in each period using optimal values computed by the first algorithm as inputs. Finally, the proposed production-inventory-queue model is implemented in a case study in PAKSHOO Chemicals Company and results are obtained and analyzed. Moreover, solving this problem can help to strategic decision making about supply chain decoupling point.

1. Introduction

Consumable products contain groups of standard goods with small volume and value per unit (e.g. foods, juices, detergents, office accessories, etc.). An ordinary consumable product manufacturer makes several products which are all the same in technological aspects. Consumers expect to find their chosen brands on the supermarket shelf anytime they go for shopping and if it is not available they probably will change their minds and buy another brand. It is because of small differences between the consumable products of different brands. In other words, in such industries demand cannot be backlogged but can be totally or partly lost. Therefore, companies of this group use MTS production policy. As the life cycle of such standard products will usually take several years, an efficient database can be prepared for forecasting future demands but about standard consumable products, companies should focus on service level and price, as the market is absolutely competitive.

In consumable product industry, production system is MTS or push policy but in chemical industries according to the competitive nature of market and expiry date of products demand forecasting and pull policies are more and more considerable. Also because of the expensive machinery and factory equipments, the optimal usage of production capacity in these industries is the main managerial worries.

As in these days competition has been extended from companies to supply chains, in this article a production-inventory planning model has been suggested with an objective function of minimizing the supply chain's total cost. Two main factors for replenishing a warehouse are delivery and cost, which should be checked out while ordering the lots. Logistic costs are the biggest

* Corresponding author.
E-mail address: mfathi@iust.ac.ir (M. Fathi).

0278-6125/$ – see front matter
doi:10.1016/j.jmsy.2010.08.003

part of supply chain costs and coordination of supply, transportation, production and inventory control as main logistic processes can cause a significant saving in total costs, on the other hand, will play an important role in achieving higher customer service levels.

The studied supply chain is a real supply chain in chemical detergent industry and we wish to plan its processes under real world's uncertain demand situations which will lead to make decisions of the inventory level, reorder quantity and production amount in every period. The proposed model is a dynamic model made of a combination of two separate models, one is an inventory control system and the other is a production planning model, therefore we had to review different parts of the literature.

An outline of the remainder of this paper is as follows. In Section 2, a literature review of related studies is presented. Section 3, describes the mathematical formulation of model. In Section 4, we state the proposed algorithm of finding optimal solutions for the problem. Finally, the computational results of PAKSHOO Chemicals Company are given in Section 5 to show the effectiveness of the developed method.

2. Literature review

2.1. Supply chain planning under uncertainty

Managing uncertainty is a main challenge within the supply chain management. The complex nature and dynamics of the relationships amongst the different actors imply an important grade of uncertainty in the planning decisions [1]. According to [2], uncertainty is defined as the difference between the amount of information required to execute a task and the information that is actually available. In SC planning decision processes, uncertainty is a key factor that can influence the effectiveness of the configuration and coordination of supply chains [3] and tends to propagate up and down along the SC, appreciably affecting its performance [1].

By the majority of studies the source of uncertainty is classified into three groups: demand, process/manufacturing and supply (e.g. [4,5]). Uncertainty in supply is caused by the variability brought about by how the supplier operates because of the faults or delays in the supplier's deliveries. Uncertainty in the process is a result of the poorly reliable production process due to, for example, machine holdups. Finally, demand uncertainty, according to Davis [6], is the most important of the three and is presented as a volatility demand or as inexact forecasting demands. In this context, it is important to highlight the works by Dejonckheere et al. [7], Disney et al. [8] and Gaalman and Disney [9] in order to measure and avoid the bullwhip effect in supply chains. The analytical models are robust optimization, stochastic programming, games theory, linear programming and parametric programming [10]. McDonald and Karimi [11] devised a mixed integer linear programming model (multi-site, multi-product, multi-period) for a mid-term SC production planning. The model they developed is of deterministic nature and adopts safety stocks to face demand uncertainties. Gupta and Maranas [12] devised a stochastic two-stage programming model based on the mixed integer linear programming model proposed by McDonald and Karimi [11] for the tactical planning of a multi-site SC with demand uncertainty. Subsequently, Gupta et al. [13] incorporated constraints to measure customer satisfaction. Gupta and Maranas [14] generalized their approach to consider the tactical planning of a multi-site, multi-product and multi-period SC with demand uncertainty. They compared the objective of finding the equilibrium between the level of customer service and the costs associated with the planning. Yu and Li [15] presented a robust optimization model based on Mulvey et al. [16] and on Mulvey and Ruszczynski [17]. It included the classic programming techniques per objective and considered scenarios to solve stochastic logistic problems. Lario et al. [18] described the generation and scenario analysis process as a tool for SC management with uncertainty in the following settings: manufacturing, assembly, distribution and service in the car manufacturing and assembly sector, within a European project framework. Lucas et al. [19] addressed the problem of planning capacities within the SC through stochastic programming with scenarios to deal with demand uncertainty. The authors applied the Lagrangian relaxation to obtain feasible integer solutions.

Nagar and Jain [20] presented a multi-stage planning model consists of supply, production and distribution under uncertain demand. They have developed a multi-period model for new products so that the demand is uncertain. The proposed model determines optimum order quantity, production quantity, distribution system and the needed outsourcings in the case of demand shortages.

Queueing theory has been extensively adopted to analyze a variety of performance analysis problems of manufacturing systems (see [21,22]). Queueing models, in turn, can be categorized as descriptive (provide values for performance measures of interest for a given configuration) or prescriptive (provide guidelines for running the system most effectively). Govil and Fu [21] conducted a comprehensive survey on queueing models for manufacturing applications. To our knowledge no supply chain planning model has been designed using queueing theory approaches and as mentioned before analytical models are usually based on robust optimization, stochastic programming, games theory, linear programming and parametric programming. In all these methods although inventory quantities are considered, warehouses have inferior roles and stochastic parameters affect all parts of planning such as supply, inventory, production, transportation, etc. but in this article using continuous review inventory system we will focus on warehouses to overcome demand uncertainty. We will use queueing theory techniques in this case.

2.2. Inventory control models with queueing theory approaches

In recent years customer service has attracted more and more attention and product availability, especially in consumable product industry, has been the most important aspect of service. This can be controlled by inventory systems; a way to present a higher service level is to classify the customers. In fact all customers of a single product do not require the same service level or do not have the same stock out fine. This inventory controlling approach was of no interest in inventory management systems and does not appear in reviewing articles [23,24], nor in basic references of supply chain and inventory control (e.g. [25,26]). The first study of demand classifications in inventory control models was done by Veinott [27]. He presented a periodic review inventory system with several demand groups and no lead time. After him a few more research were carried out by considering various customer classes in periodic inventory systems. Kleijn and Dekker [28] provided a review of inventory systems with different customer classes. By the improvement of inventory systems to continuous review policies, analyzing the usage of different demand classes approach in new systems opened new doors to researchers which were a result of information technology advances. The first work on this new approach was by Nahmias and Demmy [29]. They modeled an (s, Q) inventory system with two customer groups. Each group's demand comes under a Poisson process, unsatisfied demand is postponed. Ha [30] studied a lot-for-lot model with previous assumptions but they had considered a fixed lead time while in this model an exponentially distributed lead time was modeled; he also showed that this problem can be formulated as a queueing model. The results of Nahmias and Demmy [29] were extended by Moon and Kang [31] to a compound Poisson demand while other assumptions

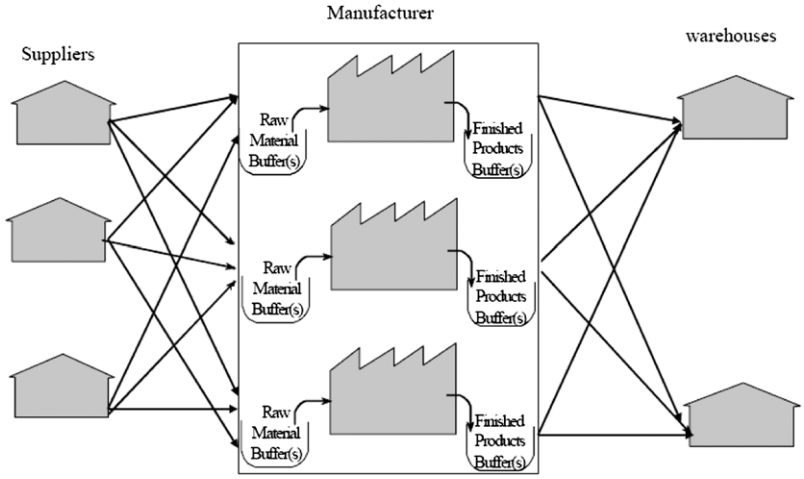


Fig. 1. Supply chain construction.

were constant. Dekker et al. [32] also presented a lot-for-lot model with the same characteristics as [29] and approximated the service level. Deshpande et al. [33] studied critical level rationing policy for demand classes which are classified by required service levels. Four control approaches have been analyzed in their study: priority fulfilling of backlogs, threshold fulfilling of backlogs, hybrid policy (combination of using threshold fulfilling parameters and the priority policy for backlog fulfilling) and optimal rationing. By their examples it is proved that, the hybrid policy works better than the threshold fulfilling and priority fulfilling policies. For small setup costs and different penalty costs for demand classes the hybrid policy cannot give significant results but for high setup costs both hybrid and the optimal rationing policy perform same as each other.

Against all these assumptions while considering lost sales, we can mention Ha [30] who modeled a single product MTS production policy with different demands coming under the Poisson process and exponentially distributed lead times. He formulated the model as an $M/M/1/S$ queue. Dekker et al. [32] as explained before presented the same model but as an $M/M/S/S$ queue. Melchiors et al. [34] provided an (s, Q) inventory system with two customer groups considering lost sales and deterministic lead times. They expressed the total expected cost formula and suggested an optimization algorithm which used some convexity characteristics to calculate optimum reorder level. Isotupa [35] developed a model with the same assumptions but exponentially distributed lead times which makes the expected cost function pseudo-convex in both parameters s and Q. In fact it is a single-item continuous review inventory system with two independent Poisson demands of different priorities to minimize replenishing, lost sales and inventory costs. She also described a computationally efficient algorithm to determine s, Q, and the optimal cost. In current article we have extended her model to multi-item supply chain with capacitated warehouse.

3. Statement of the problem

The supply chain studied is made of several suppliers, several manufacturers and several distributors which are shown in Fig. 1. The main objective is to find optimum production lot and inventory level so that the total cost of supply chain is minimized. Solving this problem can help to strategic decision making about SC decoupling point. In the studied supply chain each product can just be produced in their respective plant because of the technological availability and hygienic standards that need to be followed, so the assignment is not in our scope. Finished products will be stored in warehouses and demands fulfilled from there. Although market demands are uncertain it will be calculated and announced by distributors as production forecasts which are the Poisson processes and certain demand will be known at the beginning of each production period. Planning horizon contains $\boldsymbol{N}$ periods. Unsatisfied demand in each period is lost. In this supply chain two demand classes are considered, one is the domestic market and the other is the international market (export) which is called as higher priority market by managerial decisions. Each of these demands comes as a Poisson process with different independent rates and the lead time for the production is exponentially distributed with parameter μ (>0).

The problem will be followed in two phases: first phase is the distributor's inventory management by queueing techniques which will lead to finding safety stock and reorder quantity of each product by minimizing total cost which is structured of inventory cost, production cost and lost sales loss. The production variable cost for each product is independent of others because of BOM differences. Using different formulation causes constant production cost as well. In the second phase, a production planning problem is constructed to calculate the production quantity of each period. The scope of this study is only the finished product warehouses and manufacturing plants, not the product distribution and raw material supply.

Warehouse capacity is restricted to number of products and is not partitioned according to customers nor product group so that the whole warehouse capacity can be assigned to one kind of product. Production capacity is also restricted by available time, which is allocated to different products considering setup times and variable production times.

4. Problem formulation

This section is dedicated to mathematical formulation of model. The model is developed in two phases: inventory-queue model (Section 4.1) and production planning model (Section 4.2).

4.1. Inventory-queue model

Two demand classes are considered, one is the domestic market (distribution companies) and the other is the international market (export department) which has priority over the first one. Each of these demands comes as a Poisson process with different independent rates. Priority demand (export department) of product p arrives according to the Poisson Process with rate for $\lambda_{1p}(>0)$. Domestic demand (distribution companies) of product p arrives according to the Poisson Process with rate for $\lambda_{2p}(>0)$. As soon as the inventory level of product p reaches the safety level s_p, an order for Q_p units is placed. Therefore the maximum inventory

level is $Q_p + s_p$. The condition $Q_p > s_p$ ensures that there are no perpetual shortages. If $Q_p \le s_p$ and the inventory level reaches zero then the system will be in shortage forever. The production lead time follows an exponentially distributed function with parameter μ_p (>0). As orders are usually placed when stock levels get low, domestic demands which arrive when an order is pending are not served and hence these demands are lost. Also, when the inventory level is zero, demands due to both types of customers are assumed to be lost.

Let $I_p(t)$ denote the on-hand inventory level at time t. Since $Q_p > s_p$, at any given point of time there is at most one order pending, and as such from our assumptions it is clear that the inventory level process $\{I_p(t);\ t \ge 0\}$ with state space $E_p = \{0, 1, 2, \ldots, Q_p + s_p\}$ is a Markov process. Let

$$P_p(i, j, t) = Pr[I_p(t) = j | I_p(0) = i] \quad i, j \in E_p \tag{1}$$

$$P_p(j) = \lim_{t \longrightarrow \infty} P_p(i, j, t). \tag{2}$$

By the Markov process properties we have the following balance equations:

$$(\lambda_{1p} + \lambda_{2p}) P_p(Q_p + s_p) = \mu_p P_p(s_p) \tag{3}$$

$$(\lambda_{1p} + \lambda_{2p}) P_p(j) = (\lambda_{1p} + \lambda_{2p}) P_p(j+1) + \mu_p P_p(j - Q_p) \quad Q_p \le j \le Q_p + s_p - 1 \tag{4}$$

$$(\lambda_{1p} + \lambda_{2p}) P_p(j) = (\lambda_{1p} + \lambda_{2p}) P_p(j+1) \quad s_p + 1 \le j \le Q_p - 1 \tag{5}$$

$$(\lambda_{1p} + \mu_p) P_p(s_p) = (\lambda_{1p} + \lambda_{2p}) P_p(s_p + 1) \tag{6}$$

$$(\lambda_{1p} + \mu_p) P_p(j) = \lambda_{1p} P_p(j+1) \quad 1 \le j \le s_p - 1 \tag{7}$$

$$\mu_p P_p(0) = \lambda_{1p} P_p(1). \tag{8}$$

Solving balance equations (3)–(8) results:

$$P_p(j) = \left(1 + \frac{\mu_p}{\lambda_{1p}}\right)^{j-1} \frac{\mu_p}{\lambda_{1p}} P_p(0) \quad 1 \le j \le s_p \tag{9}$$

$$P_p(j) = \left(1 + \frac{\mu_p}{\lambda_{1p}}\right)^{s_p} \frac{\mu_p}{\lambda_{1p} + \lambda_{2p}} P_p(0) \quad s_p + 1 \le j \le Q_p \tag{10}$$

$$P_p(j) = \left[\left(1 + \frac{\mu_p}{\lambda_{1p}}\right)^{s_p} - \left(1 + \frac{\mu_p}{\lambda_{2p}}\right)^{j - Q_p - 1}\right] \frac{\mu_p}{\lambda_{1p} + \lambda_{2p}} P_p(0) \quad Q_p + 1 \le j \le Q_p + s_p. \tag{11}$$

As $\sum_{j=0}^{Q_p + s_p} P_p(j) = 1$ from Eqs. (9)–(11) we have

$$P_p(0) = \frac{\lambda_{1p} + \lambda_{2p}}{\lambda_{1p} + (\lambda_{2p} + Q_p \mu_p)\left(1 + \frac{\mu_p}{\lambda_{1p}}\right)^{s_p}}. \tag{12}$$

Now we calculate the inventory level of each product $\bar{I}_p$ as below:

$$\bar{I}_p = \left(1 + \frac{\mu_p}{\lambda_{1p}}\right)^{s_p} \left[\frac{s_p \lambda_{2p}}{\lambda_{1p} + \lambda_{2p}} + \frac{\mu_p Q_p (Q_p + 2s_p + 1)}{2(\lambda_{1p} + \lambda_{2p})} - \frac{Q_p \lambda_{1p}}{\lambda_{1p} + \lambda_{2p}} - \frac{\lambda_{1p} \lambda_{2p}}{\mu_p (\lambda_{1p} + \lambda_{2p})}\right] P_p(0) + \left(Q_p + \frac{\lambda_{2p}}{\mu_p}\right)\left(\frac{\lambda_{1p}}{\lambda_{1p} + \lambda_{2p}}\right) P_p(0). \tag{13}$$

The mean reorder rate R_p, and the mean shortage rates for the export and domestic customers per product Γ_{1p} and Γ_{2p}, are given by

$$R_p = (\lambda_{1p} + \lambda_{2p}) P_p(s_p + 1) = \mu_p \left(1 + \frac{\mu_p}{\lambda_{1p}}\right)^{s_p} P_p(0) \tag{14}$$

$$\Gamma_{1p} = \lambda_{1p} P_p(0) \tag{15}$$

$$\Gamma_{2p} = \lambda_{2p} \sum_{j=0}^{s_p} P_p(j) = \lambda_{2p} \left(1 + \frac{\mu_p}{\lambda_{1p}}\right)^{s_p} P_p(0). \tag{16}$$

We will use the notation below as well as the parameters defined above:

K_p: Setup cost of product p.
C_p: Variable production cost per unit of product p.
g_{1p}: Cost per unit shortage of product p for priority demand.
g_{2p}: Cost per unit shortage of product p for domestic demand.
h_p: Inventory holding cost of product p per unit time.

Also $C_p < g_{1p}, g_{2p}$ as by definition g_{1p}, g_{2p} are lost sales losses which mean the decrease of total revenue of supply chain. In fact this problem can be used when a fine should be paid for shortage.

The expected cost structure for product p is

$$C_p(S_p, Q_p) = h_p \bar{I}_p + (K_p + C_p Q_p) R_p + g_{1p} \Gamma_{1p} + g_{2p} \Gamma_{2p}. \tag{17}$$

Therefore the total expected cost for all products is

$$C(s, Q) = \sum_p C_p(s_p, Q_p). \tag{18}$$

As we know if $f(x_i) \ge 0$ then

$$Minimize\left(\sum_i f(X_i)\right) = \sum_i Minimize\, f(X_i). \tag{19}$$

Considering the product's independence we can solve the problem for each product independently. By substituting Γ_{2p}, Γ_{1p}, R_p, $\bar{I}_p$ in $C_p(S_p, Q_p)$, we have

$$\begin{aligned} C_p(s_p, Q_p) = {} & h_p \left(Q_p + \frac{\lambda_{2p}}{\mu_p}\right)\left(\frac{\lambda_{1p}}{\lambda_{1p} + \lambda_{2p}}\right) P_p(0) \\ & + (K_p + C_p Q_p) \mu_p \left(1 + \frac{\mu_p}{\lambda_{1p}}\right)^{s_p} P_p(0) \\ & + h_p \left(1 + \frac{\mu_p}{\lambda_{1p}}\right)^{s_p} \left[\frac{s_p \lambda_{2p}}{\lambda_{1p} + \lambda_{2p}} - \frac{\lambda_{1p} \lambda_{2p}}{\mu_p (\lambda_{1p} + \lambda_{2p})} \right. \\ & \left. - \frac{Q_p \lambda_{1p}}{\lambda_{1p} + \lambda_{2p}} + \frac{\mu_p Q_p (Q_p + 2s_p + 1)}{2(\lambda_{1p} + \lambda_{2p})}\right] P_p(0) \\ & + g_{2p} \lambda_{2p} \left(1 + \frac{\mu_p}{\lambda_{1p}}\right)^{s_p} P_p(0) + g_{1p} \lambda_{1p} P_p(0). \end{aligned} \tag{20}$$

4.2. Production planning model

A mixed integer MCLSS model is constructed here to plan the rest of the processes in supply chain. The used notation is defined below:

Sets and Indices

t : periods' index $t \in \{1, \ldots, T\}$
p : products' index $p \in \{1, \ldots, N\}$.

Decision variables

x_{pt}: The quantity of product p produced at period t
y_{pt}: $\begin{cases} 1 & \text{if product } p \text{ is produced at period } t \\ 0 & \text{otherwise} \end{cases}$
r_{pt}: The shortage for product p at period t
S_{pt}^{+}: Overstock of product p at period t
S_{pt}^{-}: Safety stock deficit of product p at period t.

Parameters

v_{pt}: Time consumption per unit of product p at period t
f_{pt}: Setup time of product p at period t
B_t: The total available capacity at period t
β_p: Price per unit of product p
D_{pt}: Demand of product p at period t.

In this model as the order quantity and rate at each period is constant – calculated by the previous model – the total demand at each period is $D_{pt} = R_p Q_p$. The inventory level of product p at period t is also defined as

$$S_{pt}^{+} + S_p - S_{pt}^{-}.$$

We assume that planning horizon contains T periods and the total number of products is N The MCLSS model formulation is represented below:

$$\text{Min} \sum_p \sum_t (c_p x_{pt} + k_p y_{pt} + g_{1p} r_{pt} + h_p(S_{pt}^{+} + s_p - S_{pt}^{-}) + g_{2p} S_{pt}^{-}) \quad (21)$$

St :

$$S_{p,t-1}^{+} - S_{p,t-1}^{-} + r_{pt} + x_{pt} = D_{pt} + S_{pt}^{+} - S_{pt}^{-}, \quad \forall p, t \quad (22)$$

$$\sum_p (v_{pt} x_{pt} + f_{pt} y_{pt}) \le B_t, \quad \forall t \quad (23)$$

$$x_{pt} \le M_{pt} y_{pt}, \quad \forall p, t \quad (24)$$

$$r_{pt} \le D_{pt}, \quad \forall p, t \quad (25)$$

$$S_{pt}^{-} \le s_p, \quad \forall p, t \quad (26)$$

$$x_{pt}, r_{pt}, S_{pt}^{+}, S_{pt}^{-} \ge 0, \quad \forall p, t \quad (27)$$

$$y_{pt} \in \{0, 1\}, \quad \forall p, t. \quad (28)$$

The objective function minimizes the total cost of production plan, that is, production costs, inventory costs, shortage costs, safety stock deficit costs and setup costs. Constraints (22) are the inventory flow conservation equations through the planning horizon. Constraints (23) are the capacity constraints; the overall consumption must remain lower than or equal to the available capacity. If we produce a product p at period t, then constraints (24) impose that the quantity produced must not exceed a maximum production level $M_{pt} \cdot M_{pt}$ can be defined as below:

$$Min \left\{ \sum_t R_p Q_p, \frac{B_t - f_{pt}}{v_{pt}} \right\} \quad (29)$$

it is the minimum value between the total demand for product p during periods $[t, T]$ of the horizon and the maximum possible production quantity according to the plant capacity. Constraints (25) and (26) define upper bounds on, respectively, the demand shortage and the safety stock deficit for product p at period t. Constraints (27) and (28) characterize the variable's domains.

5. Solution method

Before going through algorithm steps in order to reduce the computational time required finding the optimal values of Q and s, we mention the following three theorems which are based on pseudo-convexity properties.

Theorem 1. *For a fixed Q, the expected cost rate is pseudo-convex in s (See the proof in [35]).*

Theorem 2. *Whenever $Q > [\frac{(g_1 - c)\lambda_1(\lambda_1 + \lambda_2)}{h\mu}]$, $C(s, Q)$ is an increasing function of s and hence $s^* = 0$ (See the proof in [35]).*

Theorem 3. *For a fixed s, the long-run expected cost rate is pseudo-convex in Q (See the proof in [35]).*

Finding optimum s, Q algorithm steps are:

Step 1. Find $Q_{\min} = \frac{(g_1 - c)\lambda_1(\lambda_1 + \lambda_2)}{h\mu}$.

Step 2. Find s_0 and Q_0 so that

$$C(s_0, Q_0) \le C(s, Q) \quad 0 < s < Q,\ s < Q < Q_{\min}.$$

To find optimum s for each $Q < Q_{\min}$ solve the equation below:

$$C(s + 1, Q) - C(s, Q) = 0.$$

Step 3. find Q_1 so that

$$C(0, Q_1) \le -C(0, Q), \quad Q_{\min} \le Q,\ Q_1 \le \infty.$$

To find optimum $Q_{\min} \le Q$ that minimizes total costs solve $C(0, Q + 1) - C(0, Q) = 0$.

Step 4. if $C(0, Q_1) < C(s_0, Q_0)$ then $Q^* = Q_1$, $s^* = 0$ and end.

Step 5. if $C(0, Q_1) \ge C(s_0, Q_0)$ then $s^* = s_0$, $Q^* = Q_0$.

Step 6. stop.

Considering a capacitated inventory system, after computing s, Q for all products we should check the following condition: (W = *total warehouse capacity*)

$$\sum_p \bar{I}_p \le W. \quad (30)$$

If the condition is not satisfied, to calculate new s, Q we go through the following steps:

Step 1. For each product p determine below values

$$C(s_p^*, Q_p^* - 1) - C(s_p^*, Q_p^*),$$
$$C(s_p^* - 1, Q_p^*) - C(s_p^*, Q_p^*).$$

Step 2. Find the minimum among the results of the previous step, if the minimum was from the first equation then $Q_p^* := Q_p^* - 1$, otherwise $s_p^* := s_p^* - 1$.

Step 3. If by new values $\sum_p \bar{I}_p \le W$ stop, else go to step 1.

Suggested algorithm by Isotupa [35] using MAPLE 9 had the total computational time of 14.14 s to determine the optimal cost for one product and the computational time for the extended algorithm in this article using MATLAB 7.1 and implementing numerical calculation techniques for 27 products in a capacitated warehouse was 35.7 s, the efficient computational time makes this model an applicable algorithm for real world problems.

The proposed model in this article is a dynamic model combined of two separate models. Outputs of suggested algorithm to find s, Q are inputs of the second model which is coded using LINGO 8 optimization software

6. An illustrative case study: PAKSHOO Chemicals Company

PAKSHOO is a chemical manufacturing company which produces many different detergent products. Amidst the varied range of products we have chosen 27 products for our case study. Planning horizon in a year consists of 12 production periods. To avoid complicated calculations and a large number of iterations in computational algorithm and as all products are transported by truck, we changed the demand to truck scale, the capacity of current trucks is 640 boxes and all costs are scaled to one million. Input parameters for PAKSHOO Chemical Company are given in Table 1. Using the mean inventory level formula, the proposed algorithm can suggest the optimal warehouse capacity needed for current supply chain. According to numerical example we need a capacity of 680,000 boxes, and then by receiving the real capacity announced by the user which is 350,000 in this example, safety stock level and reorder quantity will be known. After changing these values to real scale we use them as inputs of MCLSS model in LINGO.

Table 1
Input parameters for PAKSHOO chemical company.

Product	β_p	λ_{1p}	λ_{2p}	v_{pt}	f_{pt}	K_p	C_p	g_{1p}	g_{2p}	h_p
1	76.8	5	11	53	30	3	43.776	89.549	81.408	0.288
2	107.52	41	90	53	30	3	61.286	125.368	113.971	0.288
3	69.12	3	7	53	30	3	39.398	80.594	73.267	0.288
4	69.12	7	16	53	30	3	39.398	80.594	73.267	0.288
5	46.08	5	11	43	30	2.3	26.266	53.729	48.845	0.256
6	107.52	10	23	53	30	3	61.286	125.368	113.971	0.288
7	107.52	10	23	53	30	3	61.286	125.368	113.971	0.288
8	107.52	10	23	53	30	3	61.286	125.368	113.971	0.288
9	76.8	5	11	53	30	3	43.776	89.549	81.408	0.288
10	76.8	5	11	53	30	3	43.776	89.549	81.408	0.288
11	107.52	41	90	53	30	3	61.286	125.368	113.971	0.288
12	107.52	25	56	53	30	3	61.286	125.368	113.971	0.288
13	69.12	8	19	53	30	3	39.398	80.594	73.267	0.288
14	96.768	7	15	53	30	3	55.158	112.831	102.574	0.288
15	23.04	4	8	32	30	1.8	13.133	26.865	24.422	0.224
16	69.12	22	49	53	30	3	39.398	80.594	73.267	0.288
17	46.08	6	13	43	30	2.3	26.266	53.729	48.845	0.256
18	128	8	19	64	30	2	72.96	149.248	135.68	0.384
19	46.08	3	6	43	30	2.3	26.266	53.729	48.845	0.256
20	23.04	14	31	32	30	1.8	13.133	26.865	24.422	0.224
21	46.08	17	38	43	30	2.3	26.266	53.729	48.845	0.256
22	107.52	10	23	53	30	3	61.286	125.368	113.971	0.288
23	107.52	7	13	53	30	3	61.286	125.368	113.971	0.288
24	76.8	8	14	53	30	3	43.776	89.549	81.408	0.288
25	76.8	18	32	53	30	3	43.776	89.549	81.408	0.288
26	69.12	4	9	53	30	3	39.398	80.594	73.267	0.288
27	46.08	3	6	43	30	2.3	26.266	53.729	48.845	0.256

As it is a mixed integer programming model, integer coefficients can deliver more realistic and sustainable outputs.

In the studied supply chain production capacity is measured by the available production time. In current situation according to economical policies, market demands and human resources costs, one of the production lines is working continuously (every 24 h and 7 days a week).

Also the second production line is producing just 2 days a week which capacity is counted as below:

LINE 1. 30 days per month*24 h a day *60 min an hour *60 s a minute=2592,000 s.

LINE 2. weeks a month, 2 days a week, 24 h a day, 60 min an hour, 60 s a minute = 691,200 s.

Total capacity : 691,200 + 2592,000 = 3283,200.

Outputs of inventory-queue model and production planning model are given in Tables 2 and 3, respectively.

Production quantities of each period (month) are outputs of this model. At the end, the total supply chain's profit will be calculated using the following equation.

$$\text{Profit Function} = \sum_p \sum_t (R_p Q_p - S_{pt}^- - r_{pt})\beta_p - \sum_p \sum_t (c_p x_{pt} + k_p y_{pt} + g_{1p} r_{pt} + h_p S_{pt}^+ + g_{2p} S_{pt}^-). \quad (31)$$

Total profit of the studied SC by this method is 428.269.653.590.

Computational results show the inventory capacity restrictions. Therefore we have analyzed the effects of capacity changes on total profit. As the real warehouse space is 10,000 m^2 it is obvious that we cannot analyze changes made by 1 box capacity increase, so to be more applicable we will discuss capacity changed for every 2500 m^2. Table 4 and Fig. 2 show the results.

The trend in Fig. 2 shows the possibility of profit increase by increasing the warehouse capacity, one of the main reasons of having such ascending trend is the high production capacity comparing with inventory capacity. According to Eq. (30) it can be seen that economic warehouse capacity occurs in equal condition.

Table 2
Outputs of inventory-queue model.

Product	s_p	Q_p	c_p	R_p	I_p
1	0.745	17	0.711*1000	0.929	9.549
2	4.548	143	8.1132*1000	0.905	74.938
3	0.409	12	0.4011*1000	0.823	6.789
4	0.983	23	0.92*1000	0.986	12.703
5	0.756	15	0.4289*1000	1.047	8.515
6	1.472	38	2.0453*1000	0.858	20.569
7	1.472	38	2.0453*1000	0.858	20.569
8	1.472	38	2.0453*1000	0.858	20.569
9	0.745	18	0.711*1000	0.878	10.049
10	0.745	18	0.711*1000	0.878	10.049
11	4.548	143	8.1132*1000	0.905	74.938
12	3.037	90	5.0175*1000	0.889	47.545
13	1.107	27	1.0797*1000	0.986	14.78
14	1.081	25	1.2286*1000	0.87	13.809
15	0.591	10	0.163*1000	1.167	5.846
16	2.547	64	2.8356*1000	1.093	34.178
17	0.904	18	0.509*1000	1.037	10.117
18	1.089	30	1.9915*1000	0.891	16.318
19	0.416	10	0.2419*1000	0.886	5.781
20	2.064	32	0.6076*1000	1.363	17.659
21	2.241	46	1.4704*1000	1.173	24.914
22	1.472	38	2.0453*1000	0.858	20.569
23	1.115	24	1.2399*1000	0.825	13.362
24	1.139	22	0.9766*1000	0.988	12.358
25	2.219	47	2.2169*1000	1.05	25.583
26	0.576	14	0.5208*1000	0.916	7.918
27	0.416	10	0.2419*1000	0.886	5.781

In this case we have 680,000 boxes as optimal value for warehouse capacity. We should know that capacity increase to more than 680,000 boxes will have no effect on SC profit and the total profit function might begin to decrease at this point. Increasing the warehouse capacity to different arbitrary values from the practical point of view is not applicable. Moreover, PAKSHOO has some limitations in warehouse space selection and 680,000 boxes as warehouse capacity are not applicable and we ignore it in our sensitivity analysis. Therefore, the two feasible capacities are 700,000 and 750,000 boxes more than 680,000 which the warehouse capacity of 700,000 boxes is optimal.

Table 3
Outputs of production planning model.

Product	$t = 1$	$t = 2$	$t = 3$	$t = 4$	$t = 5$	$t = 6$	$t = 7$	$t = 8$	$t = 9$	$t = 10$	$t = 11$	$t = 12$
1	15.8	15.8	15.8	15.8	15.8	15.8	15.8	15.8	15.8	15.8	15.8	15.8
2	129.4	129.4	129.4	129.4	129.4	129.4	129.4	129.4	129.4	129.4	129.4	129.4
3	19.8	0.0	9.9	19.8	0.0	19.8	0.0	19.8	0.0	9.9	19.8	0.0
4	22.7	22.7	22.7	22.7	22.7	22.7	22.7	22.7	22.7	22.7	22.7	22.7
5	15.7	15.7	15.7	15.7	15.7	15.7	15.7	15.7	15.7	15.7	15.7	15.7
6	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6
7	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6
8	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6
9	15.8	15.8	15.8	15.8	15.8	15.8	15.8	15.8	15.8	15.8	15.8	15.8
10	15.8	15.8	15.8	15.8	15.8	15.8	15.8	15.8	15.8	15.8	15.8	15.8
11	129.4	129.4	129.4	129.4	129.4	129.4	129.4	129.4	129.4	129.4	129.4	129.4
12	80.0	80.0	80.0	80.0	80.0	80.0	80.0	80.0	80.0	80.0	80.0	80.0
13	26.6	26.6	26.6	26.6	26.6	26.6	26.6	26.6	26.6	26.6	26.6	26.6
14	21.7	21.7	21.7	21.7	21.7	21.7	21.7	21.7	21.7	21.7	21.7	21.7
15	11.7	11.7	11.7	11.7	11.7	11.7	11.7	11.7	11.7	11.7	11.7	11.7
16	70.0	70.0	70.0	70.0	70.0	70.0	70.0	70.0	70.0	70.0	70.0	70.0
17	18.7	18.7	18.7	18.7	18.7	18.7	18.7	18.7	18.7	18.7	18.7	18.7
18	26.7	26.7	26.7	26.7	26.7	26.7	26.7	26.7	26.7	26.7	26.7	26.7
19	17.7	0.0	17.7	0.0	17.7	0.0	17.7	0.0	17.7	0.0	17.7	0.0
20	43.6	43.6	43.6	43.6	43.6	43.6	43.6	43.6	43.6	43.6	43.6	43.6
21	53.9	53.9	53.9	53.9	53.9	53.9	53.9	53.9	53.9	53.9	53.9	53.9
22	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6	32.6
23	19.8	19.8	19.8	19.8	19.8	19.8	19.8	19.8	19.8	19.8	19.8	19.8
24	21.7	21.7	21.7	21.7	21.7	21.7	21.7	21.7	21.7	21.7	21.7	21.7
25	49.3	49.3	49.3	49.3	49.3	49.3	49.3	49.3	49.3	49.3	49.3	49.3
26	12.8	12.8	12.8	12.8	12.8	12.8	12.8	12.8	12.8	12.8	12.8	12.8
27	8.9	17.7	0.0	17.7	0.0	17.7	0.0	17.7	0.0	17.7	0.0	8.9

Table 4
Supply chain total profit by different warehouse capacities.

Space	Warehouse capacity (box)	Revenue	Cost	Total profit
22,500	790,000	100,480,401,101	57,376,100,000	43,104,301,101
20,000	700,000	100,480,401,101	57,376,100,000	43,104,301,101
17,500	620,000	100,404,890,235	57,333,000,000	43,071,890,235
15,000	530,000	100,282,014,843	57,262,800,000	43,019,214,843
12,500	440,000	100,105,280,471	57,162,000,000	42,943,280,471
10,000	350,000	99,835,465,359	57,008,500,000	42,826,965,359

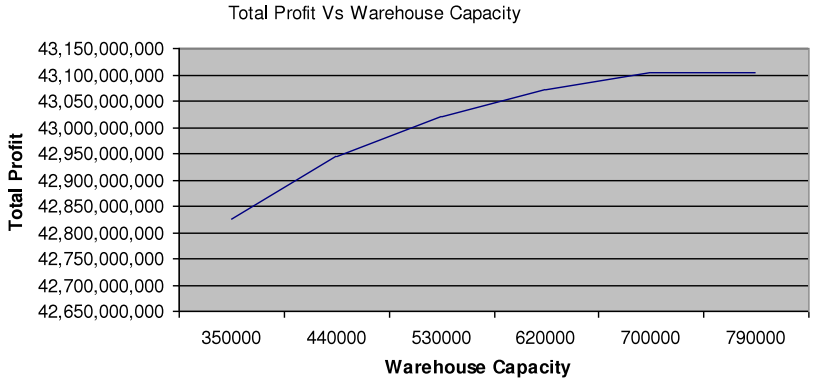


Fig. 2. Total profit vs. warehouse capacity.

7. Conclusions

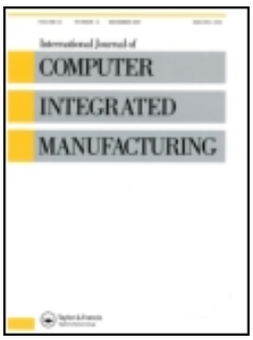
The studied supply chain process is a real time process in chemical detergent industry. We decided to plan its processes under real world's uncertain demand situations which will lead to make decisions of the inventory level, reorder quantity and production amount in every period. The proposed model is a dynamic model made with the combination of two separate models, one is an inventory control system and the other is a production planning model. To make the model more applicable we designed it under uncertain demand situations and despite common approaches such as robust optimization, stochastic programming, games theory, linear programming and parametric programming, we had to distribute all products from the central warehouse in the real situations by considering the demand uncertainty effects on production processes. It is the first time that anyone has used queueing techniques to construct this production-inventory model. Moreover, solving this problem can help to strategic decision making about supply chain decoupling point.

In this article we have extended the Isotupa [35] model to multi-item supply chain with capacitated warehouse and by implementing numerical calculation methods which decrease the computational time for efficient application in real world problems. As we combined inventory control system with production planning model for a MTS system, it is suggested that other researchers construct a similar model for MTO system or any other production system. In the second part of the model we have used LINGO to solve the formulated problem; a large-scale solution can

also be a topic of future studies. It can help to link both models more efficiently.

References

[1] Bhatnagar R, Sohal AS. Supply chain competitiveness: measuring the impact of location factors, uncertainty and manufacturing practices. Technovation 2005; 25(5):443–56.
[2] Galbraith J. Designing complex organizations. Massachussets: Addison-Wesley; 1973.
[3] Jung JY, Blau G, Pekny JF, Reklaitis GV, Eversdyk D. A simulation based optimization approach to supply chain management under demand uncertainty. Comput Chem Eng 2004;28(10):2087–106.
[4] Wang JT, Shu YF. Fuzzy decision modeling for supply chain management. Fuzzy Sets Syst 2005;150(1):107–27.
[5] Ho CF, Chi YP, Tai YM. A structural approach to measuring uncertainty in supply chains. Int J Electron Commer 2005;9(3):91–114.
[6] Davis T. Effective supply chain management. Sloan Manage Rev 1993;34(4): 35–46.
[7] Dejonckheere J, Disney SM, Lambrecht MR, Towill DR. Measuring and avoiding the bullwhip effect: a control theoretic approach. Eur J Oper Res 2003;147(3): 567–90.
[8] Disney SM, Naim MM, Potter A. Assesing the impact of e-business on supply chain dynamics. Int J Prod Econ 2004;89(2):109–18.
[9] Gaalman G, Disney SM. State space investigation of the bullwhip problem with ARMA(1, 1) demand processes. Int J Prod Econ 2006;104(2):327–39.
[10] Peidro David, Mula Josefa, Poler Raúl, Lario Francisco-Cruz. Quantitative models for supply chain planning under uncertainty: a review. Int J Adv Manuf Technol 2008;1007.
[11] McDonald CM, Karimi IA. Planning and scheduling of parallel semicontinuous processes. Production planning. Ind Eng Chem Res 1997;36(7):2691–700.
[12] Gupta A, Maranas CD. A two-stage modeling arid solution framework for multisite midterm planning under demand uncertainty. Ind Eng Chem Res 2000;39(10):3799–813.
[13] Gupta A, Maranas CD, McDonald CM. Mid-term supply chain planning under demand uncertainty: customer demand satisfaction and inventory management. Comput Chem Eng 2000;24(12):2613–21.
[14] Gupta A, Maranas CD. Managing demand uncertainty in supply chain planning. Comput Chem Eng 2003;27(8–9):1219–27.
[15] Yu CS, Li HL. A robust optimization model for stochastic logistic problems. Int J Prod Econ 2000;64(1–3):385–97.
[16] Mulvey JM, Vanderbei RJ, Zenios SA. Robust optimization of large-scale systems. Oper Res 1995;43(2):264–81.
[17] Mulvey JM, Ruszczynski AJ. A new scenario decomposition method for large-scale stochastic optimization. Oper Res 1995;43(3):477–90.
[18] Lario FC, Rodriguez A, García JP, Escudero LF. Análisis y definición de escenarios en programación estocástica para la gestión de la cadena de suministros en el sector del automóvil. IV Congreso de Ingeniería de Organización. Sevilla; 2001.
[19] Lucas C, MirHassani SA, Mitra G, Poojari CA. An application of Lagrangian relaxation to a capacity planning problem under uncertainty. J Oper Res Soc 2001;52(11):1256–66.
[20] Nagar Lokesh, Jain Karuna. Supply chain planning using multi-stage stochastic programming. Supply Chain Manag.: An Internat J 2008;13/3:251–6.
[21] Govil M, Fu M. Queueing theory in manufacturing: a survey. J Manuf Syst 1999; 18(3):214–40.
[22] Altiok T. Approximate analysis of queues in series with phase-type service and blocking. Oper Res 1989;37:601–10.
[23] Porteus EL. Stochastic inventory theory. In: Heyman DP, Sobel MJ, editors. Handbooks in OR and MS, vol. 2. Elsevier Science Publishers; 1990. North Holland.
[24] Lee HL, Nahmias S. Single-product, single-location models. In: Graves SC, Rinnooy Kan AHG, Zipkin P, editors. Handbooks in OR and MS, vol. 4. Elsevier Science Publishers; 1993. North Holland.
[25] Ballou RH. Business logistics management. 3rd ed. NJ: Englewood Cliffs; 1992. Prentice Hall.
[26] Silver EA, Pyke DF, Peterson R. Inventory management and production planning and scheduling. 3rd ed. New York: John Wiley and Sons; 1998.
[27] Veinott AF. Optimal policy in a dynamic, single product, non-stationary inventory model with several demand classes. Oper Res 1965;13:761–78.
[28] Kleijn MJ, Dekker R. An overview of inventory systems with several demand classes. Lecture Notes Econom Math 1999;480:253–65.
[29] Nahmias S, Demmy S. Operating characteristics of an inventory system with rationing. Manage Sci 1981;27:1236–45.
[30] Ha AY. Inventory rationing in a make-to-stock production system with two priority classes and backordering. Manage Sci 1997;43:1093–103.
[31] Kang MoonS. Rationing policies for some inventory systems. J Oper Res Soc 1998;49:509–18.
[32] Dekker R, Hill RM, Kleijn MJ. On the $(S-1, S)$ lost sales inventory model with priority demand classes. Naval Res Logist 2002;49(6):593–610.
[33] Deshpande V, Cohen MA, Donohue K. A threshold inventory rationing policy for service-differentiated demand classes. Manage Sci 2003;49:683–703.
[34] Melchiors P, Dekker R, Kleijn M. Inventory rationing in an (s, Q) inventory model with lost sales and two demand classes. J Oper Res Soc 2000;51(1): 11–122.
[35] Isotupa Sapna. An (s, Q) Markovian inventory system with lost sales and two demand classes. Math Comput Modelling 2006;43:687–94.



International Journal of Computer Integrated Manufacturing

ISSN: 0951-192X (Print) 1362-3052 (Online) Journal homepage: https://www.tandfonline.com/loi/tcim20

Change-point estimation of the process fraction non-conforming with a linear trend in statistical process control

F. Zandi , S.T.A. Niaki , M.A. Nayeri & M. Fathi

To cite this article: F. Zandi , S.T.A. Niaki , M.A. Nayeri & M. Fathi (2011) Change-point estimation of the process fraction non-conforming with a linear trend in statistical process control, International Journal of Computer Integrated Manufacturing, 24:10, 939-947, DOI: 10.1080/0951192X.2011.608720

To link to this article: https://doi.org/10.1080/0951192X.2011.608720

International Journal of Computer Integrated Manufacturing
Vol. 24, No. 10, October 2011, 939–947

Taylor & Francis
Taylor & Francis Group

Change-point estimation of the process fraction non-conforming with a linear trend in statistical process control

F. Zandi[a], S.T.A. Niaki[b]*, M.A. Nayeri[c] and M. Fathi[d]

[a]*Department of Industrial Engineering, K.N. Toosi University of Technology, Tehran, Iran;* [b]*Department of Industrial Engineering, Sharif University of Technology, Tehran, Iran;* [c]*Department of Industrial Engineering, Amirkabir University of Technology, Tehran, Iran;* [d]*Department of Industrial Engineering, Iran University of Science and Technology, Tehran, Iran*

(*Received 30 October 2010; final version received 6 June 2011*)

Despite the fact that control charts are able to trigger a signal when a process has changed, it does not indicate when the process change has begun. The time difference between the changing point and a signal of a control chart could cause confusions on the sources of the problems. Knowing the exact time of a process change would help to reduce the time for identification of the special cause. In this article, a model for the change-point problem is first introduced and a maximum-likelihood estimator (MLE) is applied when a linear trend disturbance is present. Then, Monte Carlo simulation is applied in order to evaluate the accuracy and the precision performances of the proposed change-point estimator. Next, the proposed estimator is compared with the MLE of the process fraction non-conforming change point derived under simple step and monotonic changes following signals from a Shewhart *np* control chart. The results show that the MLE of the process change point designed for the linear trend outperforms the MLE designed for step and monotonic changes when a linear trend disturbance is present.

Keywords: change point; process fraction non-conforming; statistical process control; process improvement; *np* charts; maximum-likelihood estimation

1. Introduction and literature review

Statistical process control (SPC) has played an important role in industry for many years. The control chart is a powerful SPC tool that monitors the changes and discovers variation in a process in order to distinguish between special and common causes of variation. In SPC, upper and lower control limits can be defined based on the probability distribution of the product's quality characteristics.

When the sample observations of the process are placed within the control limits, it can be concluded that the process is in control. However, if the sample observations are placed outside the control limits, an out-of-control signal is received. When a control chart signals an out-of-control condition, a search begins to identify and eliminate the source(s) of the special cause (see Montgomery 1996 for more details on control charts). 'The time when a special cause manifests itself into a process is referred to as change point' (Atashgar and Noorossana 2010).

Control chart's signal shows that process engineers can begin their search for the special cause of change in the process. Moreover, the disturbance in a process can be accomplished from special causes or common causes. Although control charts suggest occurrence of a change, neither can they show specific information on the cause of process disturbance, nor do they show the time of the process disturbance. In the literature of control charting methods, the change point is the time when a process begins its change by a single or multiple disturbances. However, the signalling time is the time when a control chart signals the existence of an assignable cause. Knowing the exact point of change in a process would help to search and identify special causes, resulting in time saving to find the causes. Therefore, it is useful to identify the difference between the change point and the time when an out-of-control signal is generated by control charts (Bassevile and Nikiforov 1993).

Industrial quality control setting often uses the binomial distribution to model the number of defective items in a sample of size n. Process fraction non-conforming, p, is the probability that a randomly selected item does not conform the quality characteristic. That is, given n items, the probability that x randomly selected items is defective is given by

$$P(X = x) = \binom{n}{x} p^x (1-p)^{n-x};$$
$$x = 1, 2, \ldots, n \quad 0 \le p \le 1$$

*Corresponding author. Email: Niaki@sharif.edu

ISSN 0951-192X print/ISSN 1362-3052 online

http://dx.doi.org/10.1080/0951192X.2011.608720
http://www.tandfonline.com

where p denotes the process fraction non-conforming. Depending on whether the subgroup size is constant or not, one often uses p or np charts to monitor a process. Moreover, the np, the cumulative sum (CUSUM), and the exponentially weighted moving average (EWMA) control charts are commonly used to monitor binomial counts (Ryan 2000).

Recent literatures on change-point estimation are as follows:

Hawkins and Qiu (2003) studied the change-point model for SPC. Samuel et al. (1998a, 1998b) considered step change in a normal process mean and normal process variance. Pignatielo and Samuel (2001) proposed an estimator for the change point of a normal process mean, and, based on this study, Perry and Pignatiello (2006) proposed an maximum-likelihood estimator (MLE) and evaluated the performance of this estimator when a linear trend change is present in a normal process mean. They showed that their proposed estimator provides good performance when a linear trend disturbance is present. They compared their results with suggested estimator by Samuel et al. (1998a, 1998b) for step changes. Moreover, their results showed that the MLE obtained for linear trend disturbances outperforms the MLE obtained for step change disturbances in the presence of the linear trend disturbance. Samuel and Pignatiello (1998) analysed a step change in the rate parameter for a Poisson process. Nedumaran et al. (2000) addressed the issue of change-point identification for χ^2 control chart. They used MLE to estimate a step change shift in the mean of a normal distribution. Noorossana and Shademan (2009) proposed MLE for the change point of a normal process mean that does not require the knowledge of the exact change type showed by the process. The only required assumption is that the change type present should belong to a family of monotonic change, either isotonic or antitonic. Furthermore, they compared performances between their estimator and those suggested by Samuel et al. (1998a, 1998b) and Perry and Pignatiello (2006) following a genuine signal from the Shewhart $\overline{X}$ control chart. Noorossana et al. (2009) proposed an estimator for a period of time in which a step change in the process non-conformity proportion in high-yield processes occurs. Gazanfari et al. (2008) used clustering approach to identify the time of a step change in the Shewhart control charts.

Samuel and Pignatiello (2001) proposed a MLE for the process fraction non-conforming change point by applying the step change likelihood function. They evaluated the performances of their proposed estimator when a np chart signals and concluded that their estimator provides good accuracy and precision performances. Moreover, Perry et al. (2007) developed a change-point estimator from the change likelihood function for a binomial random variable without assuming the previous information of the exact change type. The only assumption in this research is that the predicted change type is belonging to a family of monotonic change type. Further, Perry et al. (2007) compared the performances between their estimator and the one suggested by Samuel and Pignatiello (2001). In this article, a MLE is proposed for the change point of the process fraction non-conforming using the change likelihood function for a linear trend disturbance. The proposed estimator can be used for the detection of a change point when either p or np chart has shown a signal. In their research, Monte Carlo simulation is used to evaluate performances of their estimator to the commonly used MLE for the time of step change and monotonic change when a linear trend disturbance is presented following a signal from a Shewhart np control chart.

In the current research work, the change-point problem of a process fraction non-conforming is first introduced and a MLE is applied when a linear trend disturbance is present. Examples of manufacturing processes in which special causes can happen due to the linear trend disturbances in fraction non-conforming involve gradual tool wear, machine depreciation, workers' fatigue, filters that become dirty over time, or any other time-related factors that can affect the quality of produced items. Manufacturing environments with high-quality products are also some examples, in which both the fraction non-conforming and its slope of change must be low. Then, Monte Carlo simulation is applied in order to evaluate the accuracy and the precision performances of the proposed change-point estimator. Next, the proposed estimator is compared with the MLE of the process fraction non-conforming change point derived under simple step and monotonic changes following the signals from a Shewhart np control chart.

The outline of this article is as follows. We study a model for disturbance in the process when a linear trend is present in Section 2. In Section 3, we evaluate and compare the precision and the accuracy performances of the estimator. Finally, we give some concluding remarks in Section 4.

2. Linear trend change model and MLE derivation

Consider a linear trend change model for the behaviour of a process fraction non-conforming p. It is assumed the process is initially in control for the first τ subgroups and independent observations are coming

from a binomial distribution with in-control parameter $p = p_0$. Following an unknown point in time τ (the process change point), the first disturbance in the process fraction non-conforming happens. After this time, the process changes from $p = p_0$ to an out-of-control state p (where $p = p_i$; $i = \tau + 1, \tau + 2, \ldots, T$, and T denotes the time when a control chart generates a signal. A signal can be obtained when a point is either plotted above the upper control limit or an out-of-control pattern is detected using the Western Electric or other sensitising rules). Assuming the signal is not a false alarm, the change model of p is given by Equation (1), where β is the slope of the linear trend disturbance or the magnitude of process change.

$$p_i = p_0 + \beta(i - \tau) \tag{1}$$

In the proposed linear trend change model, each observation consists of a subgroup from the output of the process. For subgroups $i = 1, 2, \ldots, \tau$, the process is in control and the process fraction non-conforming is the known p_0. However, for subgroups $i = \tau + 1, \tau + 2, \ldots, T$, the process fraction non-conforming is some unknown $p_i = p_0 + \beta(i - \tau)$, where T is the most recent subgroup sample, i.e. the chart signals a change in p at subgroup number T. This model has two unknown parameters τ and β. The parameter τ represents the last subgroup taken from the in-control process, and β is the slope parameter of the linear trend model. The value of $\beta > 0$ denotes a linear change with an additive trend in p, while $\beta < 0$ represents a descending trend in the process fraction non-conforming. Based on these assumptions, the MLE can be derived for the process change point τ with non-decreasing change type ($\beta > 0$). The MLE change-point estimator is denoted by $\hat{\tau}_{lt}$.

Considering the model in Equation (1) and the above assumptions, and that the first change point takes place at time τ, the likelihood function becomes

$$L(\tau, \beta D) = \prod_{i=1}^{\tau} \binom{n}{x_i} p_0^{D_i}(1 - p_0)^{n - D_i} \prod_{i=\tau+1}^{T} \binom{n}{x_i} \times p_i^{D_i}(1 - p_i)^{n - D_i} \tag{2}$$

where n is the size of the subgroup (the subgroups size is constant) and D_i denotes the number of non-conforming units in the ith subgroup. Then, $p_i = D_i/n$ shows an estimate to the subgroup fraction non-conforming.

The MLE of τ is the value of τ that maximises the likelihood function (Equation (2)), or equivalently, its logarithm. The logarithm of the likelihood function is

$$\log_e(L(\tau, \beta \mid D)) = k + (\log_e p_0) \sum_{i=1}^{\tau} D_i + (\log_e(1 - p_0)) \times \sum_{i=1}^{\tau} (n - D_i) + \sum_{i=\tau+1}^{T} D_i \times (\log_e p_i) + \sum_{i=\tau+1}^{T} (n - D_i) \times (\log_e(1 - p_i)) \tag{3}$$

where k is a predefined constant.

Since the slope of change, β, is unknown, by taking the partial derivative of Equation (3) with regard to β and equating it zero, a formula is derived for β in terms of τ that provides the maximum value for the logarithm of the likelihood function. In other words,

$$\frac{\partial}{\partial \beta} \log_e(L(\tau, \beta \mid D)) = \sum_{i=\tau+1}^{T} \frac{D_i(i - \tau)}{p_0 + \beta(i - \tau)} - \sum_{i=\tau+1}^{T} \frac{(n - D_i) \times (i - \tau)}{1 - p_0 - \beta(i - \tau)} \tag{4}$$

Since it is difficult to find the exact values of τ and β from Equation (4) analytically, we apply the Newton method (see Hildebrand 1987) to solve Equation (4). The optimal combination of (τ,β) obtained by the Newton method is known as the MLE of the change point. This MLE of the change point can be applied when any process fraction non-conforming control chart, including CUSUM, EWMA, and *np*, gives an out-of-control signal.

In the next section, Monte Carlo simulation is used to evaluate the accuracy and the precision performances of the proposed change-point estimator following a signal from a *np* chart.

3. Performance comparison analyses

The performances of the proposed estimator, $\hat{\tau}_{lt}$, are compared with the ones of a MLE derived for step changes proposed by Samuel and Pignatiello (2001) and the one of a MLE derived for monotonic changes suggested by Perry et al. (2007) when a linear trend disturbance is present, and the out-of-control signal comes from a Shewhart *np* control chart. The estimator proposed by Samuel and Pignatiello (2001) is derived under a step change assumption, and the estimator proposed by Perry et al. (2007) is derived under a monotonic change assumption. These are referred to $\hat{\tau}_{SC}$ and $\hat{\tau}_{MC}$, respectively.

3.1. False alarms

In this section, we address the handling of false alarms in the simulation model. A signal time greater than the

real process change point, i.e. $T > \tau$, is referred to a genuine signal and can be used for searching the change point. Otherwise, however, if the signal time is less than the real change point, i.e. $T < \tau$, then the control chart signals before a disturbance in the process and, hence, is treated a false alarm.

In the simulation runs, the false alarm signal is not considered for the performance analysis. Whenever a signal is a false alarm, the process is assumed in control and, therefore, the control chart continues its action to monitor the process. In other words, when a false alarm happens in a simulation run at subgroup T, the control chart resumes at subgroup $T + 1$ while not altering the change-point estimation process. This is the identical approach used by Perry et al. (2007), Noorossana and Shademan (2009), and Perry and Pignatiello (2006).

3.2. Limitations of the chart parameters and the linear trend model

The linear trend model given in Equation (1) has some limitations. First, since it is proposed to model the process fraction non-conformities $0 \leq p \leq 1$, and, hence (Montgomery 1996),

$$0 \leq p_i = p_0 + \beta(i - \tau) \leq 1 \tag{5}$$

Then, for performance comparisons of the estimators in the presence of a linear trend disturbance, β values must be chosen such that Equation (5) holds. The change in the process fraction non-conformities along with its constraint is depicted in Figure 1.

Since only genuine alarms are considered, we have $T - \tau \geq 0$. Moreover, $p_0 \geq 0$ and $p_i \geq 0$; $i = 1, 2, \ldots$. Then, based on Figure 1, the value of T' can be obtained as follows:

$$p_T = p_0 + \beta\left(T^{'} - \tau\right) = 1 \Rightarrow T^{'} = \tau + \frac{1 - p_0}{\beta} \tag{6}$$

Now, since $T' \geq T$ is considered, β must have values such that Equation (6) on T' holds.

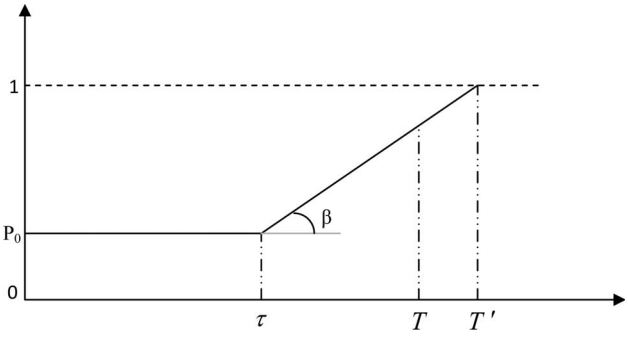


Figure 1. The constraint on the process fraction non-conforming ($0 \leq p \leq 1$).

Second, it was mentioned in the previous section that the *np* chart is used to monitor the process. Since the number of non-conforming items in each subgroup cannot be negative, i.e. $D_i \geq 0$, the lower control limit (LCL) cannot be negative either. Thus, the minimum number of each subgroup in the simulation runs is set such that Equation (7) holds.

$$\mathrm{LCL} = np_0 - 3\sqrt{np_0(1 - p_0)} \geq 0 \tag{7}$$

3.3. Performances of the MLEs for $p_0 = 0.01$ and $n = 300$

In this section, Monte Carlo simulation is used to evaluate the performances (i.e. the bias and the variability of the estimates on τ) of the change-point estimators for an in-control fraction non-conforming of $p_0 = 0.01$ with a subgroup size of $n = 300$, where the process real change point is simulated to happen at $\tau = 50$. Independent observations for simulation model are sampled from a binomial distribution with the process fraction non-conforming $p_0 = 0.01$ and subgroups of $i = 1, 2, \ldots, 50$, each having a size of $n = 300$. After subgroup 50, independent observations are simulated from a binomial distribution with the process fraction non-conforming $p_i = p_0 + \beta(i - 50)$ until the control chart signals. Based on the simulation data, the three aforementioned estimators of the process fraction non-conforming change point, i.e. $\hat{\tau}_{lt}$, $\hat{\tau}_{\mathrm{SC}}$, and $\hat{\tau}_{\mathrm{MC}}$, were then obtained. This procedure was repeated $N = 10{,}000$ times over a range of β values for each estimator. The mean squared errors, MSE, and the expected time of the change-point estimates were calculated as shown in Table 1, where the standard errors greater or equal than 0.01 are shown in parentheses. Moreover, the expected time of the first genuine alarm, $E(T)$, is the expected time at which the control chart first signals a disturbance in the process fraction non-conforming.

The results provided in Table 1 show that, except for $\beta = 0.01$, $\mathrm{MSE}(\hat{\tau}_{\mathrm{lt}})$ is smaller than both $\mathrm{MSE}(\hat{\tau}_{\mathrm{SC}})$ and $\mathrm{MSE}(\hat{\tau}_{\mathrm{MC}})$ for all other considered values of β. It means that the proposed estimator performs better than the other two estimators. Note that, in this table, as the magnitude of the slope parameter, β, increases to 0.30, the mean squared error for the three estimators decreases. However, more accurate estimates are obtained using the proposed method in almost all cases. Thus, it can be concluded from Table 1 that the proposed estimator outperforms the other two estimators and that it provides a more accurate estimate of the true process change point when a linear trend disturbance is present.

In order to evaluate and compare the precision of proposed change-point estimator with the ones of the estimators proposed by Samuel and Pignatiello (2001) and Perry et al. (2007), this procedure was repeated for a total of $N = 10{,}000$ independent simulation runs using $p_0 = 0.01$, $n = 300$, and $\tau = 50$ for each estimator. Then, the probability of the change-point estimate to lie within a certain sample from the true change point is reported in Table 2 for different values of β. In this table, the precision estimates of $\hat{\tau}_{MC}$ are shown in

Table 1. Accuracy performances for three MLEs of the change point for different β values following a genuine signal from a *np* control chart when a linear trend change is present.

β	$E(T)$	$\hat{\tau}_{lt}$	MSE($\hat{\tau}_{lt}$)	$\hat{\tau}_{MC}$	MSE($\hat{\tau}_{MC}$)	$\hat{\tau}_{SC}$	MSE($\hat{\tau}_{SC}$)
0.01	52.189	51.167 (0.01)	1.361	38.742 (0.27)	126.804	49.413 (0.06)	0.348
0.02	51.391	50.320 (0.01)	0.102	37.967 (0.28)	144.871	49.196 (0.06)	0.648
0.03	51.090	50.007	0.004	38.335 (0.28)	136.137	49.088 (0.06)	0.834
0.04	50.984	49.835	0.027	38.882 (0.27)	123.669	49.169 (0.05)	0.693
0.05	50.950	49.665	0.112	39.039 (0.36)	120.277	49.346 (0.05)	0.429
0.07	50.949	49.669	0.109	37.742 (0.40)	150.422	49.299 (0.05)	0.493
0.08	50.947	49.755	0.050	38.007 (0.39)	143.983	49.338 (0.05)	0.440
0.09	50.950	49.809	0.036	38.582 (0.39)	130.523	49.261 (0.05)	0.548
0.11	50.943	49.888	0.012	39.040 (0.36)	120.255	49.312 (0.05)	0.476
0.13	50.949	49.933	0.004	38.306 (0.39)	136.903	49.337 (0.05)	0.441
0.15	50.949	49.947	0.003	37.712 (0.40)	151.158	49.247 (0.05)	0.569
0.19	50.947	49.910	0.007	38.487 (0.39)	132.690	49.357 (0.05)	0.415
0.23	50.946	49.565	0.189	38.453 (0.38)	133.477	49.2642 (0.05)	0.543
0.25	50.946	49.324	0.456	39.000 (0.38)	121.135	49.259 (0.05)	0.551
0.30	50.938	49.625 (0.01)	0.141	37.214 (0.41)	163.648	49.452 (0.09)	0.308

Table 2. Precision performance of the three estimators based on different values of β ($p_0 = 0.01$, $n = 300$, $\tau = 50$, and $N = 10{,}000$ independent runs).

β	0.01	0.02	0.03	0.04	0.05	0.07	0.09	0.11	0.13	0.15	0.19	0.23	0.25	0.30
$P(\lvert T-\tau\rvert = 0)$	0.35	0.48	0.62	0.71	0.79	0.89	0.93	0.94	0.92	0.95	0.95	0.95	0.95	0.95
	(0.02)	(0.02)	(0.01)	(0.01)	(0.01)	(0.01)	(0.01)	(0.02)	(0.02)	(0.01)	(0.01)	(0.01)	(0.01)	(0.01)
	[0.21]	[0.57]	[0.82]	[0.78]	[0.84]	[0.86]	[0.90]	[0.91]	[0.94]	[0.95]	[0.95]	[0.87]	[0.86]	[0.51]
$P(\lvert T-\tau\rvert \le 1)$	0.45	0.90	0.92	0.92	0.93	0.94	0.94	0.94	0.94	0.94	0.94	0.95	0.95	.95
	(0.10)	(0.09)	(0.09)	(0.10)	(0.09)	(0.09)	(0.11)	(0.09)	(0.10)	(0.09)	(0.09)	(0.09)	(0.10)	(0.07)
	[0.76]	[0.99]	[1.00]	[1.00]	[1.00]	[1.00]	[0.99]	[0.99]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]
$P(\lvert T-\tau\rvert \le 2)$	0.73	0.92	0.93	0.93	0.94	0.95	0.95	0.95	0.95	0.95	0.95	0.96	0.96	0.96
	(0.21)	(0.20)	(0.20)	(0.23)	(0.20)	(0.22)	(0.22)	(0.19)	(0.21)	(0.21)	(0.21)	(0.20)	(0.21)	(0.17)
	[0.96]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]
$P(\lvert T-\tau\rvert \le 3)$	0.79	0.94	0.94	0.95	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.97	0.97	0.97
	(0.30)	(0.28)	(0.29)	(0.31)	(0.29)	(0.29)	(0.30)	(0.29)	(0.29)	(0.29)	(0.29)	(0.28)	(0.29)	(0.25)
	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	1.00]	[1.00]	[1.00]	[1.00]
$P(\lvert T-\tau\rvert \le 5)$	0.86	0.95	0.95	0.96	0.96	0.97	0.97	0.97	0.97	0.97	0.97	0.98	0.98	0.98
	(0.44)	(0.38)	(0.41)	(0.44)	(0.41)	(0.40)	(0.43)	(0.42)	(0.43)	(0.21)	(0.42)	(0.41)	(0.43)	(0.38)
	[1.00]	[1.00]	[1.00]	[1.00	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]
$P(\lvert T-\tau\rvert \le 7)$	0.91	0.95	0.96	0.96	0.96	0.97	0.97	0.97	0.98	0.98	0.98	0.98	0.98	0.98
	(0.54)	(0.51)	(0.54)	(0.56)	(0.54)	(0.51)	(0.55)	(0.55)	(0.54)	(0.31)	(0.54)	(0.54	(0.55)	(0.49)
	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]
$P(\lvert T-\tau\rvert \le 9)$	0.94	0.95	0.96	0.96	0.96	0.97	0.97	0.97	0.98	0.98	0.98	0.98	0.98	0.98
	(0.64)	(0.61)	(0.69)	(0.65)	(0.64)	(0.1)	(0.64)	(0.65)	(0.62)	(0.59)	(0.63)	(0.63)	(0.64)	(0.60)
	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]
$P(\lvert T-\tau\rvert \le 13)$	0.95	0.95	0.96	0.96	0.96	0.97	0.97	0.98	0.98	0.98	0.98	0.99	0.98	0.8
	(0.73)	(0.71)	(0.73)	(0.73)	(0.75)	(0.72)	(0.73)	(0.75)	(0.72)	(0.60)	(0.74)	(0.72)	(0.74)	(0.70)
	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]
$P(\lvert T-\tau\rvert \le 17)$	0.96	0.95	0.96	0.96	0.96	0.97	0.98	0.98	0.98	0.98	0.98	0.99	0.99	0.99
	(0.78)	(0.78)	(0.78)	(0.79)	(0.80)	(0.77)	(0.79)	(0.80)	(0.77)	(0.76)	(0.79)	(0.78)	(0.80)	(0.75)
	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]
$P(\lvert T-\tau\rvert \le 19)$	0.96	0.95	0.96	0.96	0.96	0.97	0.98	0.98	0.98	0.98	0.98	0.99	0.99	0.99
	(0.82)	(0.80)	(0.81)	(0.81)	(0.83)	(0.78)	(0.81)	(0.83)	(0.81)	(0.78)	(0.81)	(0.80)	(0.83)	(0.77)
	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[.00]	[1.00]	[1.00]	[1.00]
$P(\lvert T-\tau\rvert \le 20)$	0.96	0.95	0.96	0.96	0.96	0.97	0.98	0.98	0.98	0.98	0.98	0.99	0.99	0.99
	(0.83)	(0.81)	(0.81)	(0.82)	(0.84)	(0.79)	(0.81)	(0.84)	(0.82)	(0.79)	(0.81)	(0.81)	(0.84)	(0.79)
	[1.00]	[1.00]	[1.00]	[1.00]	1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]	[1.00]

parentheses and the precision estimates of $\hat{\tau}_{lt}$ are shown in brackets.

Based on the results in Table 2, the $\hat{\tau}_{lt}$ estimator provides a more or at least equal precise estimate in comparison with $\hat{\tau}_{SC}$ and $\hat{\tau}_{MC}$ if the changes follow a linear trend model. Moreover, regarding to the results obtained in Table 2, the precision estimates of the estimators are plotted in Figures 2–7, where they show precision estimate values versus possible slope of change trends for specified tolerances. Further, for each value of β, three values of different estimators are compared. In these figures, the precision performances of the proposed estimator are shown in comparison with the other estimators in the presence of linear disturbance.

Figure 2 shows that, for process change with $\beta = 0.03$, the estimated probability of correctly identifying the time of the process change using the proposed estimator is 82%, whereas the estimated probabilities of $\hat{\tau}_{SC}$ and $\hat{\tau}_{MC}$ are 62% and 1%, respectively. It should be noted that the precision provided by the proposed estimator of the accurate change point (limiting value of the probability $P(|T - \tau| = 0)$ was not absolutely better than $\hat{\tau}_{SC}$, where, for some values of β, the precision performance of $\hat{\tau}_{SC}$ is equal or even better than the ones of the proposed estimator. For example, the estimated limiting values for this

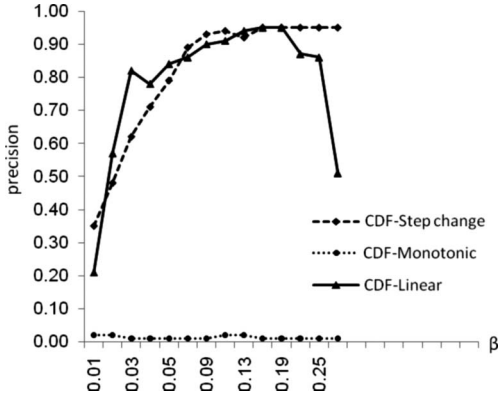


Figure 2. Precision of estimators for the estimated accurate change point $P(|T - \tau| = 0)$.

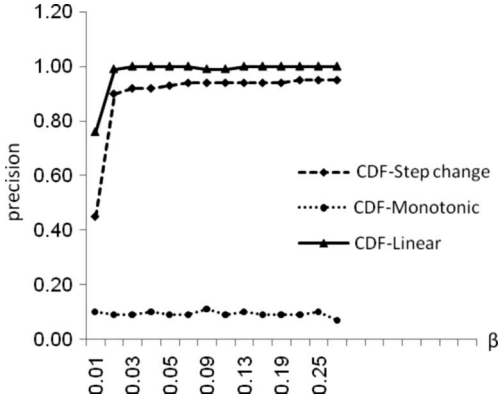


Figure 3. Precision of estimators for tolerance 1 subgroup $P(|T - \tau| \leq 1)$.

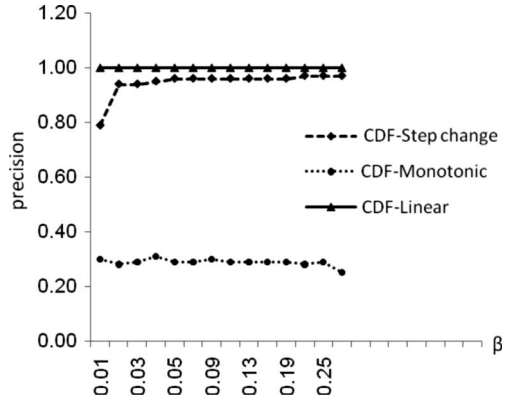


Figure 4. Precision of estimators for tolerance 3 subgroups $P(|T - \tau| \leq 3)$.

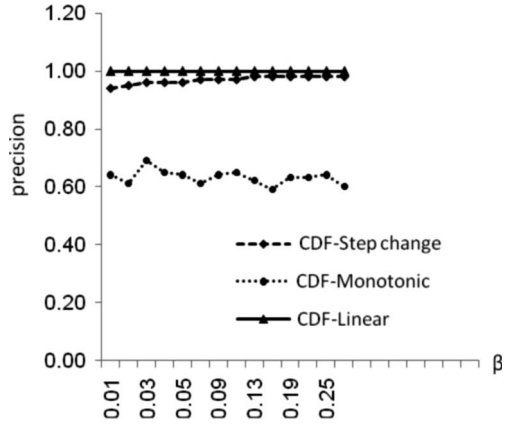


Figure 5. Precision of estimators for tolerance 9 subgroups $P(|T - \tau| \leq 9)$.

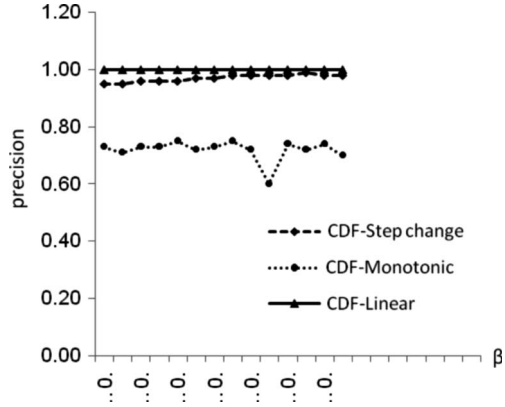


Figure 6. Precision of estimators for tolerance 13 subgroups $P(|T - \tau| \leq 13)$.

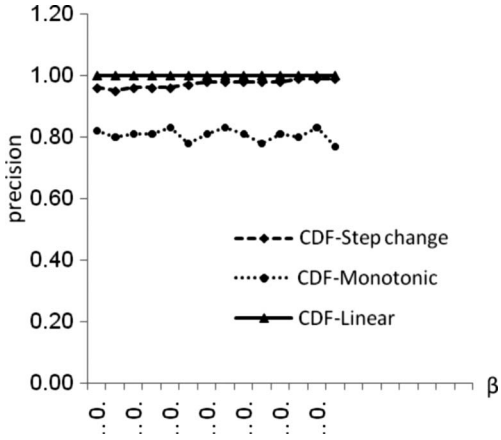


Figure 7. Precision of estimators for tolerance 17 subgroups $P(|T - \tau| \leq 17)$.

probability with change slope of $\beta = 0.11$, using $\hat{\tau}_{SC}$ and $\hat{\tau}_{lt}$, are 0.82 and 0.49, respectively. However, for other limiting values of the probability, $\hat{\tau}_{SC}$ has perfect precision.

The precision provided by the proposed estimator for the true change point within 1 subgroup, which is in the presence of linear disturbance, is better than the other two estimators. While increasing the magnitude of the change slope in the process fraction non-conforming, it is observed that the precision preference of the proposed estimator is as good as that of the accurate performance as shown in Table 1. Moreover, the precision of $\hat{\tau}_{SC}$ is improved by the increase in β values. As observed in Figures 2–7, by increasing the tolerance value of the precision, $\hat{\tau}_{MC}$ has a rapid trend, never having an acceptable precision in comparison with the other two estimators. Moreover, it can be seen that the estimation of the change point using the proposed estimator is within 3 subgroups of the true process change point in all the simulation runs, whereas the other two estimators do not have such precision within 20 subgroups of the true process change point.

3.3.1. *The parameter constraint*

In this section, the parameter constraints (Equations (5) and (7)) that was mentioned in Section 3.2 are considered, where the process change point is simulated to happen at $\tau = 50$. Independent observations are simulated from a binomial distribution the with process fraction non-conforming of $p_0 = 0.01$ and subgroup size of $n = 300$. Using Equation (6), the value of T' is obtained as

$$T' = 50 + \frac{1 - 0.01}{\beta} \tag{8}$$

Table 3. Limitation of the β value.

β	$E(T)$	$\hat{\tau}_{lt}$	Standard error	$MSE(\hat{\tau}_{lt})$	Count
1.50	46.866	39.00	0.00	1.00	8661
2.00	46.864	39.00	0.00	1.00	8641
2.50	46.866	39.00	0.00	1.00	8661

Table 3 shows the simulation results for ineligible β values that have no suitable circumstances for comparison between estimators.

A *count* variable enumerates the simulation runs that the control chart signal time, T, is greater than the calculated T'. If the *count* value of the simulation runs is considerable (for example, greater than or equal to 50) for each β value, then this β value is not suitable for the comparison between the estimators. The above results are repeatable for other estimators as well. With control charts that are more precise relative to the *np* chart, like CUSUM and EWMA, the accuracy, and the precision performance analysis of the estimating change-point methods can be improved.

The second constraint is concerned on the size of the subgroups such that the LCL becomes non-negative. Regarding this constraint, for the specific simulation runs at hand, the minimum number of a subgroup can be determined using Equation (7) as

$$\begin{aligned} \text{LCL} &= n \times 0.01 - 3\sqrt{n \times (0.01) \times (1 - 0.01)} \geq 0 \\ &\Rightarrow n \geq 300 \end{aligned} \tag{9}$$

In other words, the minimum subgroup size required for the *np* chart to have non-negative LCL is 300. Samuel and Pignatiello (2001) have also indicated that as the size of the subgroup increases, the accuracy and the precision performance of the estimator improve. It is up to the process engineering to determine the subgroup size considering economic limitation, lower limitation, and the minimum precision.

It should be noted that although the batches are mostly made small today, there are still many manufacturing processes that make large batches. Moreover, in the proposed methodology, one can make the batches smaller and obtain negative LCL that can be assumed zero.

4. Conclusion

When a control chart signals an out-of-control condition, a search begins to identify and, hence, to eliminate the source(s) of the special cause. The time

when a special cause manifests itself into a process is referred to a change point. Estimation of the genuine time and the real source of the disturbance cause(s) in the process fraction non-conforming is valuable for process engineers and technicians who would like to gain more ease and quick identification of the variables and/or procedures that might cause a change in their processes. In this article, an estimator based on the maximum likelihood was proposed that helps to identify the change point when a disturbance of linear nature shifts the process fraction non-conforming. Manufacturing processes in which special causes can happen due to the linear trend disturbances in fraction non-conforming involve gradual tool wear, machine depreciation, workers' fatigue, filters that become dirty over time, or any other time-related factors that can affect the quality of produced items. Moreover, the performance of the proposed method was compared with the ones of two other available estimators that were developed by Samuel and Pignatiello (2001) and Perry et al. (2007) in the presence of step change and monotonic change type, respectively. The results of this research showed that the MLE obtained for the linear trend change has better performance than the ones for the step change and monotonic change type when a linear trend disturbance is present. We note that if a linear process change is simulated, the model that proposes a linear process change will give the best results. In practice, at the appearance of the out-of-control signal, one does not know the mathematical function of process disturbance. Usually, when the out-of-control signal is detected, the diagnostic is started. One of the diagnoses can suppose a linear process change. In this case, the proposed method is useful in SPC.

The following ideas may be considered for future research:

(1) Using more accurate methods, like CUSUM and EWMA, to monitor the process fraction non-conforming may result in better estimates of the process change point.
(2) In addition to the 'above upper control limit' rule that causes an out-of-control signal, Western Electric or other sensitising rules may also be employed to improve the precision of the future change-point estimator.
(3) The proposed estimator has been compared with the step change and monotonic change type estimators when a linear trend disturbance is present. In this case, we showed the proposed estimator outperforms the other two estimators available in the literature. The comparison study for other kinds of practical changes, for example, step or periodic disturbance, may be investigated in future.
(4) While the proposed methodology has been tested using simulation, finding a real manufacturing case study may be considered in the future.

Acknowledgements

The authors are thankful for the constructive comments of the reviewers and the editor that certainly improved the presentation of the article.

References

Atashgar, K. and Noorossana, R., 2010. An integrating approach to root cause analysis of a bivariate mean vector with a linear trend disturbance. *International Journal of Advanced Manufacturing Technology*. doi: 10.1007/s00170–010–2728-x.

Bassevile, M. and Nikiforov, I.V., 1993. *Detection of abrupt changes: theory and applications*. New Jersey: Prentice Hall.

Gazanfari, M., *et al*., 2008. A clustering approach to identify the time of a step change in Shewhart control charts. *Quality and Reliability Engineering International*, 24 (7), 765–778.

Hawkins, D.M., and Qiu, P., 2003. The change point model for statistical process control. *Journal of Quality Technology*, 35 (4), 355–366.

Hildebrand, F.B., 1987. *Introduction to numerical analysis*. 2nd ed. New York: Dover Publications (reprinted 1987).

Montgomery, D.C., 1996. *Introduction to statistical quality control*. 3rd ed. New York: John Wiley & Sons, Inc.

Nedumaran, G., Pignatiello, J.J. Jr., and Calvin, J.A., 2000. Identifying the time of a step-change with χ^2 control charts. *Quality Engineering*, 13 (2), 153–159.

Noorossana, R., and Shademan, A., 2009. Estimating the change point of a normal process mean with a monotonic change. *Quality and Reliability Engineering International*, 25 (1), 79–90.

Noorossana, R., *et al*., 2009. Identifying the period of a step change in high-yield processes. *Quality and Reliability Engineering International*, 25 (7), 875–883.

Perry, M.B., and Pignatiello, J.J. Jr., 2006. Estimation of the change point of a normal process mean with a linear trend disturbance. *Quality Technology and Quantitative Management*, 3 (3), 325–334.

Perry, M.B., Pignatiello, J.J. Jr., and Simpson, J.R., 2007. Estimating of the change point of the process fraction nonconforming with a monotonic change disturbance in SPC. *Quality and Reliability Engineering International*, 23 (3), 327–339.

Pignatielo, J.J. Jr., and Samuel, T.R., 2001. Estimation of the change point of a normal process mean in SPC applications. *Journal of Quality Technology*, 33 (1), 82–95.

Ryan, T.P., 2000. *Statistical methods for quality improvement*. New York: Wiley.

Samuel, T.R. and Pignatiello, J.J. Jr., 1998. Identifying the time of a step change in a Poisson rate parameter. *Quality Engineering*, 10 (4), 673–681.

Samuel, T.R. and Pignatiello, J.J. Jr., 2001. Identifying the time of a step change in the process fraction nonconforming. *Quality Engineering*, 13, 357–365.

Samuel, T.R., Pignatiello, J.J. Jr., and Calvin, J.A., 1998a. Identifying the time of a step change with XBar control charts. *Quality Engineering*, 10 (3), 521–527.

Samuel, T.R., Pignatiello, J.J. Jr., and Calvin, J.A., 1998b. Identifying the time of a step change in a normal process variance. *Quality Engineering*, 10 (3), 529–538.

Int J Adv Manuf Technol (2011) 57:235–244
DOI 10.1007/s00170-011-3261-2

ORIGINAL ARTICLE

Price, delivery time, and capacity decisions in an M/M/1 make-to-order/service system with segmented market

E. Teimoury · M. Modarres · A. Kazeruni Monfared · M. Fathi

Received: 18 May 2010 / Accepted: 7 March 2011 / Published online: 15 April 2011

Abstract Speed and price are the two most important factors in customer satisfaction and business success in today's competitive environment. Time-based product differentiation and segment pricing have provided firms with a great opportunity to profit enhancement. This paper presents a coding system for pricing/queuing models in the literature. In this article, a service/make-to-order firm with heterogeneous price and delivery time-sensitive customers as an M/M/1 queuing system is analyzed. The firm uses customers' heterogeneity to create market segments. Products offered to each segment differ only in price and delivery time. The objective of this profit-maximizing firm is to determine optimal price, delivery time, and capacity for different market segments. Moreover, solving this problem can help to strategic decision making about supply chain decoupling point. An approach based on uniformization and matrix geometric method so as to calculate the distribution of low-priority customers' time in system is developed. Then, the proposed pricing/queuing model is implemented by a numerical study and firm's optimal decisions under shared and dedicated capacity strategies are analyzed and the effect of capacity costs and product substitution is studied. Finally, we have shown how firm's decisions are influenced by market characteristics, capacity costs, and operational strategies.

Keywords Time-based product differentiation · Pricing · Capacity management · Delivery time guarantees · Queuing systems

E. Teimoury · A. K. Monfared · M. Fathi (✉)
Department of Industrial Engineering,
Iran University of Science and Technology,
Tehran, Iran
e-mail: mfathi@iust.ac.ir

E. Teimoury
Logistics and Supply Chain Researches and Studies Group,
Institute for Trade Studies and Research,
Tehran, Iran

M. Modarres
Department of Industrial Engineering,
Sharif University of Technology,
Tehran, Iran

1 Introduction

An effective way to maintain customer responsiveness and to enhance demand is through time-based product differentiation and segment pricing (Boyaci and Ray [4]). Offering products which are different only in delivery times and prices is a common strategy in markets with heterogeneous price and time-sensitive customers. Many companies, especially in service and make-to-order (MTO) industries, are using delivery time guarantees as their marketing strategy. Since shorter delivery times allow firms a price premium, they try to benefit from customers' sensitivity to speed. Adopting time-based product differentiation brings capacity-related issues to the forefront because speed of product delivery is directly influenced by a firm's capacity.

Companies that offer different products should decide whether a given product is available to all customers or distinct groups. For example, FedEx quotes different logistic services with different guaranteed delivery times to every customer willing to pay its price. Customers select the appropriate option based on their preferences for speed and willingness to pay. In this case, the menu of products offered is substitutable and demands are dependent. On the other hand, the price and delivery time combinations that Dell offers to government and health care corporations are

different from those offered to individuals. Dell decides on a product's availability to each market segment, and customers have no choice. In this case, options are non-substitutable, and the demand from each segment is independent from others.

As a firm's marketing decisions are closely linked to its operation strategies, time-based product differentiation would have a direct influence on operational systems that produce and deliver these products. A natural question that comes to mind is whether firms differentiate systems used for production and delivery of different products or not. In other words, companies should decide on using shared or dedicated capacities.

The last two decades have observed significant research progress on the interaction of pricing and operations. This literature falls into two fundamental categories: pricing/inventory models and pricing/queuing models. In the first case, prices are determined jointly with inventory decisions or based on current inventory levels. In the second case, prices are used to control the arrival rate to a queue or queues and may or may not be set based on the current queue length (Ray and Jewkes [27]). Here, we have classified pricing/queuing models based on two general specifications: problem definition and modeling assumptions. A coding system is proposed in Table 1 and models available in recent literature are coded based on this system in Table 2. Among these pricing/queuing literatures, the closest works to ours are by Boyaci and Ray [4] and Sinha et al. [29]. Boyaci and Ray [4] studied a firm using dedicated facilities to serve two customer classes with different delivery time guarantees and at different prices. They modeled mean demand from each customer class as a linear function of its own price and delivery time as well as price and delivery time quoted to the other class. Our demand function is more general and uses different sensitivities (to price and time) for each class of customers. Besides, we consider two different operational strategies (shared and dedicated). Sinha et al. [29] consider a firm which uses shared capacities and delay dependent priority discipline to serve existing (primary class) and new customers (secondary class).

An outline of the remainder of this paper is as follows. In Section 2, the problem is defined more precisely. Section 3 is dedicated to the mathematical formulation of our model. Computational results are discussed in Section 4. We conclude the paper in Section 5.

2 Statement of the problem

According to the new business model of Internet/telephone ordering and quick response time requirement, MTO business model is growing quickly. We consider an MTO or a service firm that serves customers with different sensitivities to price and delivery time. The firm uses customers' heterogeneity to create market segments and offers them different prices and delivery times for the same product. For simplicity, we assume there are two classes of customers, express customers—who are more time sensitive and are willing to pay a price premium—and regular customers who are more price sensitive and are willing to accept a longer delivery time for a price discount than a shorter delivery time. Moreover, solving this problem can help strategic decision making about supply chain decou-

Table 1 Coding system proposed for classification

Problem Definition			Modeling assumptions		
Market	Monopolistic	Mo	Objective function	Profit maximization	MxP
	Competitive	Co		Value maximization	MxV
Customers	Homogeneous	Hm	Constraints	Delivery time reliability	DTR
	Heterogeneous	Ht		Customer's utility	CU
Pricing	Internal (transfer)	In		Expected waiting time	EWT
	External	Ex	Decision variables	Price	P
Operation strategy	Shared capacity	ShC		Capacity	C
	Dedicated capacity	DeC		Delivery time	D
Product differentiation	With	W		Other	O
	Without	WO	Queuing system	M/M/1	MM1
Product substitution	Substitutable	S		Other	O
	Non-substitutable	NS	Demand	Linear	L
				Nonlinear	NL
			Cost	Capacity	CC
				Operating	OC
				Delay	DC

Table 2 Classification of reviewed article

Item no.	Author(s)	Article's code (problem definition/modeling assumptions)
1	Mendelson [17]	Mo/Hm/In/DeC/SP/NS/MxP&MxV//P&C/O/L/CC&DC
2	Dewan and Mendelson [6]	Mo/Ht/In/ShC/SP/NS/MxV//P&C/MM1/L/CC&DC
3	Mendelson and Whang [18]	Mo/Ht/In/ShC/MP/NS/MxV//P&C/MM1/L/DC
4	Hill and Khosla [11]	Mo/Hm/Ex/DeC/SP/NS/MxP//P&D//NL
5	Stidham [32]	Mo/Ht/In/ShC/SP/NS/MxP//P&C/O/L/CC&DC
6	Li and Lee [15]	Co/Ht/Ex/DeC/SP/NS/MxP//P&C&O/MM1/NL
7	Lederer and Li [14]	Co/Ht/Ex/ShC/MP/NS/MxP//P&O/O/L/DC
8	So and Song [31]	Mo/Hm/Ex/DeC/SP/NS/MxP/DTR/P&D&C/MM1/NL/CC&OC
9	Palaka et al. [20]	Mo/Hm/Ex/DeC/SP/NS/MxP/DTR/P&D/MM!/L/OC&DC
10	Ha [8]	Mo/Hm/In/DeC/SP/NS/MxP&MxV//P/O/L/OC&DC
11	Rao and Petersen [26]	Mo/Ht/In/ShC/MP/NS/MxP&MxV//P/MM1/L/OC&DC
12	So [30]	Co/Hm/Ex/DeC/SP/NS/MxP/DTR/P&D/MM1/L/OC
13	Van Mieghem [34]	Mo/Ht/Ex/ShC/MP/NS/MxP//P&O/MM1/L/OC&DC
14	Ha [9]	Mo/Ht/In/ShC/MP/S/MxV//P/O/L/OC&DC
15	Hall et al. [10]	Mo/Ht/Ex/ShC/MP/NS/MxP/EWT/P/L/
16	Boyaci and Ray [4]	Mo/Ht/Ex/DeC/MP/S/MxP/DTR/P&C&D/MM1/L/CC&OC
17	Mandjes [16]	Mo/Ht/Ex/ShC/MP/S&NS/MxP//P&C/MM1/L/CC&OC
18	Ray and Jewkes [27]	Mo/Hm/Ex/DeC/SP/NS/MxP/DTR/P&D/MM1//L/CC&OC
19	Allon and Federgruen [3]	Co/Hm/Ex/DeC/SP/NS/MxP/DTR/P&D/MM1//L/CC&OC
20	Afeche [1]	Mo/Ht/Ex/ShC/MP/NS/MxP/CU/P&O/MM1/L
21	Afeche and Mendelson [2]	Mo/Ht/Ex/ShC/MP/NS/MxP&MxV//P/MM1/L/DC
22	Katta and Sethuraman [13]	Mo/Ht/Ex/ShC/MP/NS/MxP//P&O/MM1/L/DC
23	Boyaci and Ray [5]	Mo/Ht/Ex/DeC/MP/S/MxP/DTR/P&C&D/MM1/L/CC&OC
24	Pekgun et al. [23]	Mo/Hm/Ex/DeC/SP/NS/MxP/DTR/P&D/MM1/L
25	Dobson and Stavrulaki [7]	Mo/Hm/Ex/DeC/SP/NS/MxP/CU/P&C&O/O/L/CC&DC
26	Allon and Federgruen [3]	Co/Ht/Ex/DeC&ShC/MP/NS/MxP/DTR/P&D/MM1/L/CC&OC
27	Pekgun et al. [22]	Co/Hm/Ex/DeC/SP/NS/MxP/DTR/P&D/MM1/L
28	Pangburn and Stavrulaki [21]	Mo/Ht/Ex/DeC&ShC/SP/NS/MxP/CU/P&C/MM1&O/L/CC&DC
29	Sinha et al. [29]	Mo/Ht/Ex/ShC/MP/NS//MxP/EWT/P&DC&O/MM1/L

pling point. Assumptions and decisions determined by the model are explained as follows.

2.1 Assumptions

- Demand from customer class i arrives according to a Poisson process with rate λ_i that depends not only on its own price and delivery time but also on price and delivery time quoted to the other class.
- Customers cannot observe the congestion in the firm, and their choices are based on the prices and delivery times offered.
- The time taken to serve each demand from class i is exponentially distributed with rate μ_i, therefore the service facility is modeled as an M/M/1 queuing system. We also assume that in SC, both service rates are equal to μ.
- Customers within each class are served based on FCFS priority discipline.
- Applying SC strategy, express customers are served based on preemptive priority discipline.
- All customers are served by the same service capacities hence capacity costs are equal for both classes.
- The operating cost the firm incurs for serving customers of either class is equal.
- Delivery time is predetermined and fixed for regular customers.

2.2 Decisions

- Which prices are to be offered for product/service to express and for regular customers.
- Which delivery time is to be quoted to express customers.
- Using different capacity strategies, which service rates are to be used in order to meet the guaranteed delivery times with a determined service level.

We also investigate how these decisions are influenced by changes in capacity costs, capacity strategies, and product substitutability. According to the coding system

presented in Table 2, our studied problem will be coded as: Mo/Ht/Ex/DeC & ShC/W/S&NS/MxP/DTR&EWT/P&C&D/MM1/L/CC&OC.

3 Problem formulation

This section is dedicated to mathematical formulation of model. The following notations are used for the mathematical formulation of our problem:

Sets and indices

i Customer classes' index i=1,2

Parameters

$2a$ Market size
β_p^i Sensitivity of customer within class i to its own price
β_L^i Sensitivity of customer within class i to its own guaranteed delivery time
θ_p Sensitivity of demand to interclass price difference
θ_L Sensitivity of demand to interclass price difference
C Unit operating cost
A Capacity cost
α Service level

Variables

p_i Price quoted to customers within class i
L_i Delivery time guaranteed to customers within class i
μ_i Mean service rate for customers within class i
λ_i Mean arrival rate for customers within class i
T_{si} Time in system for customers within class i

The mathematical formulation of the problem is as follows:

$$\text{Max} \sum_{i=1}^{2} p_i \times \lambda_i - \sum_{i=1}^{2} c \times \lambda_i - \sum_{i=1}^{2} A \times \mu_i \tag{1}$$

Subject to

$$\text{Stability condition} \tag{2}$$

$$\Pr(T_{si} \le L_i) \ge \alpha \qquad \forall i \tag{3}$$

$$L_1 < L_2 \tag{4}$$

$$p_i \ge 0, \mu_i \ge 0, L_1 \ge 0 \tag{5}$$

Objective function (1) maximizes a firm's profit. Constraint (2) is the stability condition for the M/M/1 queuing system used for a modeling service facility. Constraint (3) imposes that the time that each customer spend in the system (time in queue+service time) of a customer should not exceed the guaranteed delivery time related to his class with a probability of at least α. Constraint (4) assures that the guaranteed delivery time for express customers be shorter than regular customers. According to assumptions in Section 2.1., demand from customer class i arrives according to a Poisson process with rate λ_i, which depends not only on its own absolute price and delivery time but also on its price and delivery time quoted relative to the other class. Then, the firm can attract new customers by reducing price and/or by offering shorter delivery times. The price and/or delivery time reduction for one class can also induce customers to switch preferences. It is assumed that customers cannot observe the congestion levels of the firms, and their choices are only based on the prices and delivery times announced by the firms. The demand rates are modeled using the linear functions (Eq. 6), inspired by Tsay and Agrawal [33], Boyaci and Ray [4]:

$$\lambda_i = \begin{cases} \lambda_1 = a - \beta_p^1 p_1 + \theta_p(p_2 - p_1) - \beta_L^1 L_1 + \theta_L(L_2 - L_1), i = 1 \\ \lambda_2 = a - \beta_p^2 p_2 + \theta_p(p_1 - p_2) - \beta_L^2 L_2 + \theta_L(L_1 - L_2), i = 2 \end{cases} \tag{6}$$

This demand model (Eq. 6) also confirms the effects of price and time differentiation on the demand rates: one extra unit of price differentiation decreases the demand rate from express customer and increases that from regular customers by the same amount, while one extra unit of time differentiation increases the demand rate from express customers and decreases that from regular customers by the same amount (Jayaswal et al. [12] and Sachin Jayaswal [28]). Constraints (2) and (3) can be written as

$$\lambda_i < \mu_i \qquad \forall i \tag{7}$$

$$\Pr(T_{si} \le L_i) = 1 - e^{(\lambda_i - \mu_i)L_i} \ge \alpha \qquad \forall i \tag{8}$$

while using dedicated capacities and as

$$\lambda_1 + \lambda_2 < \mu \tag{9}$$

$$Pr(T_{si} \le L_1) = 1 - e^{(\lambda_1 - \mu)L_i} \ge \alpha \qquad i = 1 \tag{10}$$

while using shared capacities. For regular customers, a closed form expression for distribution of time in a system does not exist. We describe the continuous time Markov chain model for an SC system in Section 3.1. Steady state probabilities are computed using the matrix geometric method (MGM). Ramaswami and Lucantoni [24] and Neuts [19] presented an algorithm based on

uniformization to derive the complimentary distribution function of the stationary waiting times in quasi birth and death (QBD) processes. We adopt their algorithm to derive T_{s2}.

3.1 The Markov chain

Consider the continuous time Markov chain $\{(i,j), i \geq 0, 0 \leq j \leq M\}$, where i (the level) and j (the interlevel state) are respectively the number of low and high priority customers in the system. The generator matrix of this QBD process Q is given as

$$Q = \begin{bmatrix} B_{0,0} & A_0 & 0 & 0 & \dots \\ A_2 & A_1 & A_0 & 0 & \dots \\ 0 & A_2 & A_1 & A_0 & \dots \\ 0 & 0 & A_2 & A_1 & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

where

$$A_0 = \begin{bmatrix} \lambda_2 & 0 & 0 & \dots \\ 0 & \lambda_2 & 0 & \dots \\ 0 & 0 & \lambda_2 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \quad A_2 = \begin{bmatrix} \mu & 0 & 0 & \dots \\ 0 & 0 & 0 & \dots \\ 0 & 0 & 0 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

$$B_{0,0} = \begin{bmatrix} -(\mu+\lambda_2) & \lambda_1 & 0 & \dots \\ \mu & -(\mu+\lambda_2+\lambda_1) & \lambda_1 & \dots \\ 0 & \mu & -(\mu+\lambda_2+\lambda_1) & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} A_1 = \begin{bmatrix} -(2\mu+\lambda_2) & \lambda_1 & 0 & \dots \\ \mu & -(\mu+\lambda_2+\lambda_1) & \lambda_1 & \dots \\ 0 & \mu & -(\mu+\lambda_2+\lambda_1) & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

Submatrices A_0, A_1, A_2, and $B_{0,0}$ show transitions from level i to level $i+1$, transitions within level $i>1$, transitions from level i to level $i-1$ and transitions within level $i=0$, respectively.

3.2 Stability conditions

Let $A = A_0 + A_1 + A_2$. We have

$$A = \begin{bmatrix} -\mu & \lambda_1 & 0 & \dots \\ \mu & -(\mu+\lambda_1) & \lambda_1 & \dots \\ 0 & \mu & -(\mu+\lambda_1) & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

It is obvious that A is a generator matrix and its associated stationary distribution $\pi = [\pi_0, \pi_1, \dots, \pi_M]$ is driven as a solution to $\pi A=0$ and $\pi 1=1$.

3.3 The stationary distribution

Let $x = [x_0, x_1, x_2, \dots]$ be the stationary distribution associated with the Markov chain such that $x1=1$ and $xQ=0$.

From the matrix geometric theorem we know that

$$x_{i+1} = x_i R, \qquad i \geq 0$$

where R is the minimal nonnegative solution to the matrix quadratic equation

$$A_0 + RA_1 + R^2 A_2 = 0$$

The matrix R can be computed very easily using the iterative approach presented by Ramaswami and Lucantoni [25] and Neuts [19].

The boundary vector x_0 is obtained from

$$x_0\left(B_{0,0} + RA_2\right) = 0$$

We then normalize it by

$$x_0(I-R)^{-1}e = 1$$

3.4 Computing regular customers' time in system

Based on the approach presented by Ramaswami and Lucantoni [24] and Neuts [19], the probability that a regular customer arriving to the system at an arbitrary time will wait longer than x units is equal to

$$\Pr(T_s > x) = \sum_{n=0}^{\infty} d_n e^{-\theta x} \frac{(\theta x)^n}{n!}$$

where $\theta = \max_{1 \leq j \leq m} -(A_0 + A_1)_{jj}$ and

$$d_n = \pi_0 (I-R)^{-1} R H_n e$$
$$H_0 = I \qquad H_{n+1} = H_n \widehat{A}_1 + R H_n \widehat{A}_2$$

where

$$\widehat{A}_1 = \frac{1}{\theta}(A_0 + A_1) + I \qquad \widehat{A}_2 = \frac{1}{\theta} A_2$$

Table 3 Values of parameters

PDSM		TDSM		β_p^1	β_p^2	β_L^1	β_L^2	A	a	c	α	L_2
θ_p	θ_L	θ_p	θ_L									
25	10	10	25	30	40	45	25	15	1000	3	0.99	3

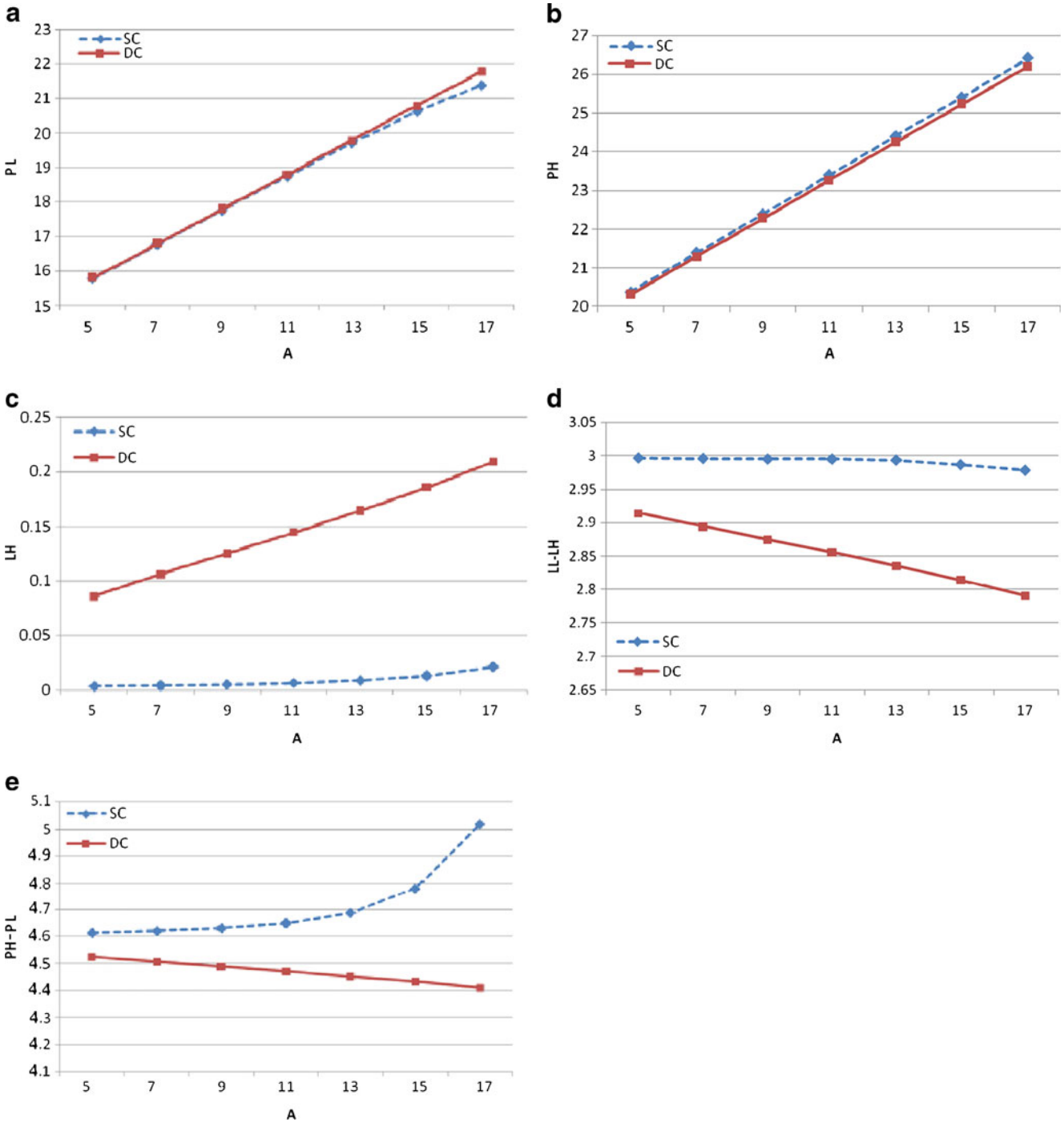


Fig. 1 The effect of capacity cost increase on **a** prices offered to regular customers, **b** prices offered to express customers, **c** delivery time offered to express customers, **d** delivery time difference, and **e** price difference, in a time difference-sensitive market

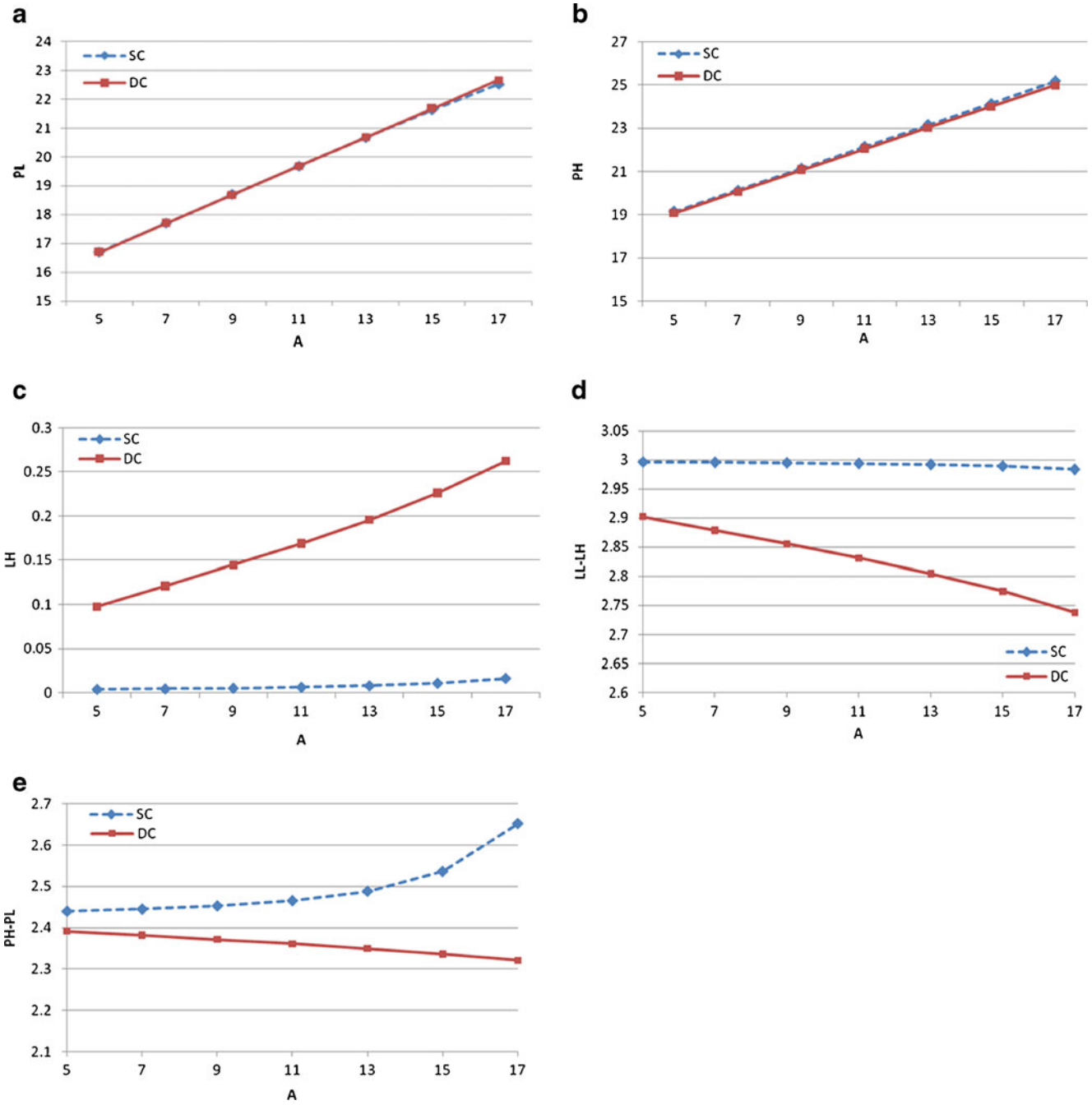


Fig. 2 The effect of capacity cost increase on **a** prices offered to regular customers, **b** prices offered to express customers, **c** delivery time offered to express customers, **d** delivery time difference, and **e** price difference, in a price difference-sensitive market

Now we can rewrite constraint (3) for regular customers as

$$\Pr(T_{si} < L_2) = 1 - \sum_{n=0}^{\infty} e^{-\theta x} \frac{(\theta x)^n}{n!} \pi_0 (I - R)^{-1} R H_n e > \alpha, \quad i = 2$$

4 Numerical examples

In this section, we illustrate how a firm's decisions relate to the capacity cost and operational strategies and present the optimal solution for a variety of parameter settings. Parameters θ_L and θ_p are assumed to be different for time difference-sensitive markets (TDSM) and price difference-sensitive markets (PDSM). Parameter values are presented in Table 3.

To study the effect of capacity costs on a firm's optimal decisions, we solved the problem for A=5, 7, 9, 11, 13, 15, 17. The results for TDSM, PDSM, and non-substitutable products (θ_p=θ_L=0) are respectively illustrated in Figs. 1, 2, and 3.

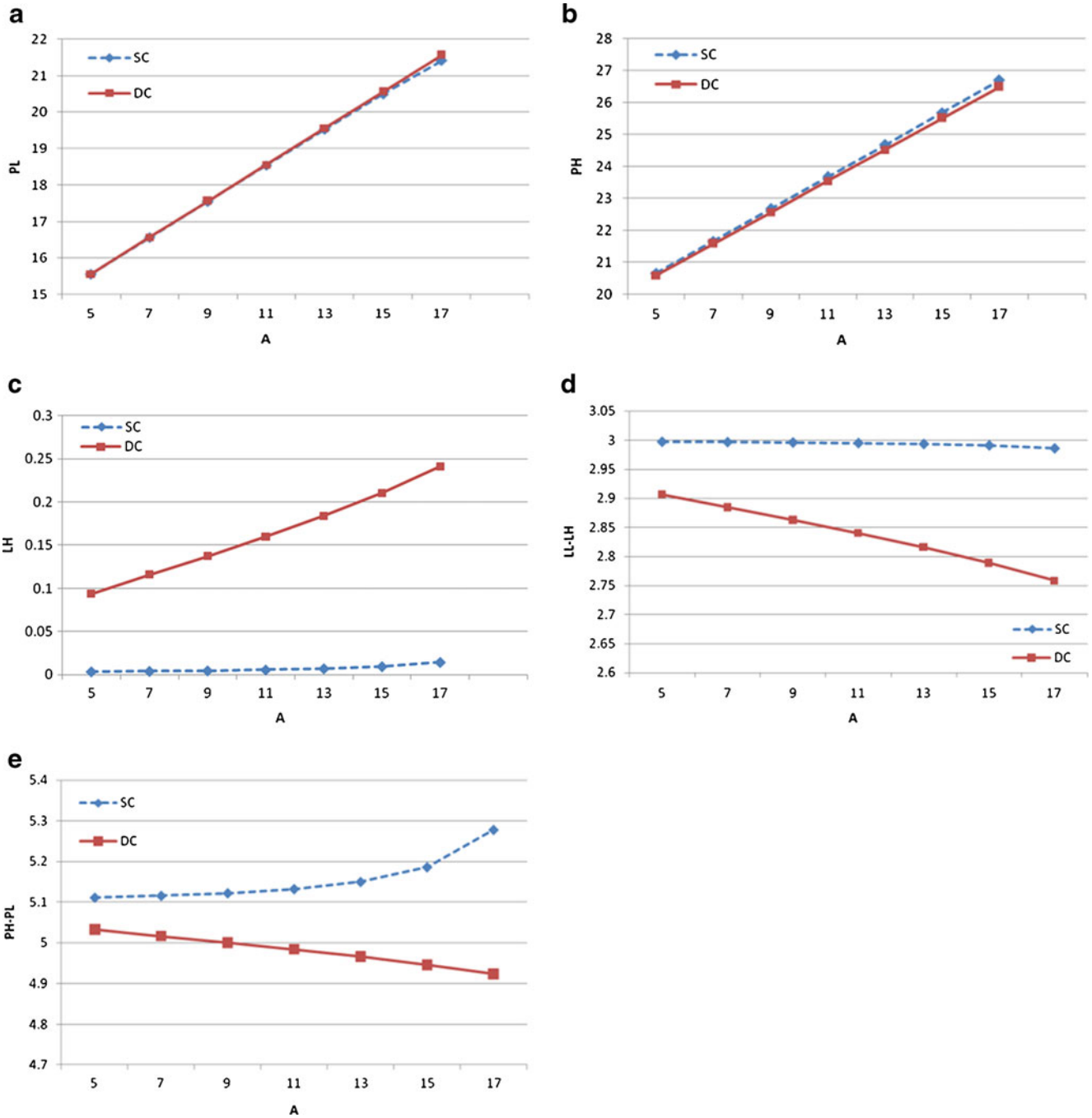


Fig. 3 The effect of capacity cost increase on **a** prices offered to regular customers, **b** prices offered to express customers, **c** delivery time offered to express customers, **d** delivery time difference, and **e** price difference for non-substitutable products

As shown above, irrespective of market characteristics, for a firm using shared capacities, the optimal decision in reaction to an increase in marginal capacity cost is to increase the delivery time for express customers and prices for both classes. Obviously, when the capacity costs increase, it will not be beneficial for the firm to expand its capacities, therefore the service rate decreases and the delivery time will increase. The delivery time and the prices should be set so that the delivery time differentiation decreases and the price differentiation increases. For a firm

Table 4 Results for $A=15$ in TDSM, PDSM, and non-substitutable cases

	TDSM		PDSM		Non-substitutable	
	SC	DC	SC	DC	SC	DC
p_1	25.12542	25.32329	23.9578	23.8629	25.45487	25.32329
p_2	20.64079	20.5625	31.05279	21.63891	20.44629	20.5625
L_1	0.284547	0.45783	0.31452	0.49139	0.06384	0.45783
μ_1	339.6329	229.7573	335.4478	240.8582	334.64511	229.7573
μ_2		104.0351		91.49204		104.0351
Profit	1,898.76	1,697.669	1,701.837	1,520.929	1,837.234	1,697.669

using dedicated capacities, optimal decisions are similar to the ones in a shared capacity strategy, but delivery time and prices should be set so that the delivery time differentiation and price differentiation decrease.

The results for $A=15$ in TDSM, PDSM, and non-substitutable cases are reported in Table 4. Tables 5 and 6 illustrate the effect of product substitution and capacity strategies on a firm's optimal decisions and profit.

Product substitution will lead to a decrease in an express product's price and an increase in a regular product's price and express product's delivery time (see Table 4). In other words, when products are substitutable, the price and delivery time differentiations are less than non-substitutable cases (see Table 4). Besides, using dedicated capacities in TDSM and PDSM, μ_1 is larger and μ_2 is smaller compared to non-substitutable case.

As presented in Table 5, using shared capacities to serve different customer classes will cause an increase in an express product's price and a decrease in a regular product's price and express product's delivery time. In other words, when capacities are shared, price and delivery time differentiations are more than dedicated cases.

5 Conclusions

According to the new business model of Internet/telephone ordering and quick response time requirement, the MTO business model is growing quickly. In this article, the authors have studied an MTO service firm that serves nonhomogeneous price and time-sensitive customers. The firm uses nonhomogeneity and differentiates its product based on the delivery time. Therefore, different delivery times and prices are offered to different customer classes. In this study, the firm modeled as an M/M/1 queuing system. Then, an approach based on uniformization and MGM so as to calculate the distribution of low-priority customers' time in a system is developed. A numerical example is provided and a firm's optimal decisions under shared and dedicated capacity strategies are analyzed. The effect of capacity costs and product substitution is studied. Consequently, solving this queuing–pricing problem can help strategic decision making about the supply chain decoupling point. This study modeled the firm as an M/M/1 queuing system. More general systems like G/G/1 and implementation in a real case study such as an electronic product supply chain can also be a topic of future studies. Moreover, considering competitive markets would be a challenging problem.

Table 5 Effect of product substitution on a firm's optimal decisions

	TDSM		PDSM	
	SC	DC	SC	DC
p_1	Decrease	Decrease	Decrease	Decrease
p_2	Increase	Increase	Increase	Increase
p_1–p_2	Decrease	Decrease	Decrease	Decrease
L_1	Increase	Increase	Increase	Increase
L_1–L_2	Decrease	Decrease	Decrease	Decrease

Table 6 Effect of using shared capacities on a firm's optimal decisions

	TDSM	PDSM	Non-substitutable
p_1	Increase	Increase	Increase
p_2	Decrease	Decrease	Decrease
p_1–p_2	Increase	Increase	Increase
L_1	Decrease	Decrease	Decrease
L_1–L_2	Increase	Increase	Increase

References

1. Afeche P (2004) Incentive compatible revenue management in queueing systems: optimal strategic delay and other delaying tactics. Working paper, Northwestern University
2. Afeche P, Mendelson H (2004) Pricing and priority auctions in queueing systems with a generalized delay cost structure. Manage Sci 50(7):869–882
3. Allon G, Federgruen A (2005) Competition in service industries. Oper Res 53(1):37–55
4. Boyaci T, Ray S (2003) Product differentiation and capacity cost interaction in time and price sensitive markets. Manuf Serv Oper Manage 5(1):18–36
5. Boyaci T, Ray S (2006) The impact of capacity costs on product differentiation in delivery time, delivery reliability, and price. Prod Oper Management POMS 15(2):179–197
6. Dewan S, Mendelson H (1990) User delay costs and internal pricing for a service facility. Manage Sci 36(12):1502–1517
7. Dobson G, Stavrulaki E (2007) Simultaneous price, location, and capacity decisions on a line of time-sensitive customers. Nav Res Logistics NRL 54(1):1–10
8. Ha AY (1998) Incentive-compatible pricing for a service facility with joint production and congestion externalities. Manage Sci 44 (12):1623–1636
9. Ha AY (2001) Optimal pricing that coordinates queues with customer chosen service requirements. Manage Sci 47(7):915–930
10. Hall JM, Kopalle PK, Pyke DF Static and dynamic pricing of excess capacity in a make-to-order environment. Working paper, Tuck School of Business at Dartmouth
11. Hill AV, Khosla IS (1992) Models for optimal lead time reduction. Prod Oper Manage 1(2):185–197
12. Jayaswal S, Jewkes E, Ray S (2011) Product differentiation and operations strategy in a capacitated environment. Eur J Oper Res 210(3):716–728
13. Katta A, Sethuraman J (2005) Pricing strategies and service differentiation in queues: a profit maximization perspective. Department of Industrial Engineering and Operations Research, Columbia University
14. Lederer PJ, Li L (1997) Pricing, production, scheduling, and delivery-time competition. Oper Res 45(3):407–420
15. Li L, Lee YS (1994) Pricing and delivery-time performance in a competitive environment. Manage Sci 40(5):633–646
16. Mandjes M (2003) Pricing strategies under heterogeneous service requirements. Comput Netw 42(2):231–249
17. Mendelson H (1985) Pricing computer services: queuing effects. Commun ACM 28(3):312–321. doi:10.1145/3166.3171
18. Mendelson H, Whang S (1990) Optimal incentive-compatible priority pricing for the M/M/1 queue. Oper Res 38(5):870–883
19. Neuts MF (1981) Matrix-geometric solutions in stochastic models: an algorithmic approach. Dover Publications, Mineola
20. Palaka K, Erlebacher S, Kropp DH (1998) Lead-time setting, capacity utilization, and pricing decisions under lead-time dependent demand. IIE Trans 30(2):151–163. doi:10.1023/A:1007414117045
21. Pangburn MS, Stavrulaki E (2008) Capacity and price setting for dispersed, time-sensitive customer segments. Eur J Oper Res 184 (3):1100–1121
22. Pekgun P, Griffin PM, Keskinocak P (2006) Centralized vs. decentralized competition for price and lead-time sensitive demand. H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta (in press)
23. Pekgun P, Griffin PM, Keskinocak P (2008) Coordination of marketing and production for price and lead time decisions. IIE Trans 40(1):12–30
24. Ramaswami V, Lucantoni DM (1985) Stationary waiting time distribution in queues with phase type service and in quasi-birth-and-death processes. Stoch Models 1(2):125–136
25. Ramaswami V, Lucantoni DM (1999) An introduction to matrix analytic methods in stochastic modeling. SIAM, Philadelphia
26. Rao S, Petersen ER (1998) Optimal pricing of priority services. Oper Res 46(1):46–56
27. Ray S, Jewkes EM (2004) Customer lead time management when both demand and price are lead time sensitive. Eur J Oper Res 153 (3):769–781
28. Jayaswal S (2009) Product differentiation and operations strategy for price and time sensitive markets. PhD Thesis, University of Waterloo
29. Sinha SK, Rangaraj N, Hemachandra N (2010) Pricing surplus server capacity for mean waiting time sensitive customers. Eur J Oper Res 205(1):159–171
30. So KC (2000) Price and time competition for service delivery. Manuf Serv Oper Manage 2(4):392–409
31. So KC, Song JS (1998) Price, delivery time guarantees and capacity selection. Eur J Oper Res 111(1):28–49
32. Stidham S (1992) Pricing and capacity decisions for a service facility: stability and multiple local optima. Manage Sci 38 (8):1121–1139
33. Tsay AA, Agrawal N (2000) Channel dynamics under price and service competition. Manuf Serv Oper Manage 2(4):372–391. doi:10.1287/msom.2.4.372.12342
34. Van Mieghem JA (2000) Price and service discrimination in queuing systems: incentive compatibility of Gcμ scheduling. Manage Sci 46 (9):1249–1267. doi:10.1287/mnsc.46.9.1249.12238

Int J Adv Manuf Technol
DOI 10.1007/s00170-012-3913-x

ORIGINAL ARTICLE

A queuing approach for making decisions about order penetration point in multiechelon supply chains

E. Teimoury · M. Modarres · I. G. Khondabi · M. Fathi

Received: 18 May 2010 / Accepted: 10 January 2012

Abstract This study is dedicated to order penetration point (OPP) strategic decision making which is the boundary between make-to-order (MTO) and make-to-stock (MTS) policies. A multiproduct multiechelon production supply chain is considered where the first production stage manufactures semifinished products based on an MTS policy to supply the second production stage which operates on the MTO policy. The producer desires to find the optimal fraction of processing time fulfilled by supplier and optimal semifinished products buffer capacity in OPP. To calculate system performance indexes, the matrix geometric method is employed. Afterward, optimal solutions are obtained by enumeration and direct search techniques. Moreover, the system behavior is analyzed by the numerical example. It is shown that system total cost is a concave function of increasing completed percentage in first production stage. According to the total cost function elements, managers desire to locate OPP where to balance the order fulfillment delay cost, holding cost and the cost of disposing unsuitable items. Finally, the impact of different amounts of storage capacity on OPP and total cost are analyzed. Also, the manner of expected numbers of unsuitable products, semifinished products, and expected order completion delay are analyzed versus various quantities of storage capacity and production rate.

Keywords Queuing system · Supply chain · Logistics · Order penetration point (OPP) · MTS/MTO queue · Matrix geometric method (MGM)

E. Teimoury · M. Fathi (✉)
Department of Industrial Engineering,
Iran University of Science and Technology,
Tehran, Iran
e-mail: mfathi@iust.ac.ir

E. Teimoury
Logistics & Supply Chain Researches & Studies Group,
Institute for Trade Studies & Research,
Tehran, Iran

M. Modarres
Department of Industrial Engineering,
Sharif University of Technology,
Tehran, Iran

I. G. Khondabi
Department of Engineering, Shahed University,
Tehran, Iran

1 Introduction

In terms of supply chain design, customer satisfaction lead time and inventory costs are common problems that keep managers actively encouraged. In a production supply chain system, there are various generators of uncertainties such as demands' arrival times, processing times, transportation lead times, reliability of the machines, and the capability of the operators [1]. Customers consider the response time of the order completion as a service performance measure [2]. According to the new business model of Internet/telephone ordering and quick response time requirement, make-to-order (MTO) business model is growing quickly [3, 4]. On the one hand, make-to-stock (MTS) production system can meet customer orders fast, but confronts inventory risks associated with short product life cycles and unpredictable demands. On the other hand, MTO producers can provide a variety of products and custom orders with lower inventory risks, although usually have longer customer lead times. Moreover, in MTS production, products are stocked in advance, while in MTO production, a product only starts to be produced when an order of demand is received. In some cases, custom products share approximately all the parts of the standard products and can be produced by alternating the existing standard parts with some further works, thus the assembler usually

Published online: 26 February 2012

contemplates embedding MTO processes into the mainstream MTS lines which in turn forms a hybrid production system. Order penetration point (OPP) is a concept which enables the decision makers to make use of a hybrid MTS/MTO system, applying the abovementioned queuing theory. This is considered a suitable way to model uncertainties which affects the OPP.

There are some articles of making decisions on OPP which appeared in the literature with many names such as decoupling point (DP), delay product differentiation (DPD), and product customization postponement. The term DP, in the logistics framework, was first introduced by Sharman [5] where he argued the DP's dependency on a balance between product cost, competitive pressure, and complexity. Adan and Van der Wal [6] analyzed the effect of MTS and MTO production policies on order satisfaction lead times. Arreola-Risa and DeCroix [7] studied a simple queuing environment where customers are served in first-come-first-served order regardless of their classes. Moreover, they provided a closed form formulae for making decision about the production strategy for each customer type. Recently, Yavuz Günalay [8] studied the efficient management of MTS or MTO production–inventory system in a multi-item manufacturing facility. Rajagopalan [9] proposed a mixed-integer nonlinear program production model which optimized (Q, r) (the production lot size and inventory reorder point), parameters of every product's inventory system. A comprehensive literature review on MTS–MTO production systems and revenue management of demand fulfillment can be found in Perona et al. [10] and Quante et al. [11]. The trade-off between aggregation of inventory (or inventory pooling) and the costs of redesigning the production process is studied by Aviv and Federgruen [12] where they do not consider congestion impacts, whereas Gupta and Benjaafar [13] considered the impact of capacity restrictions and congestion. That is, they proposed a common framework to examine MTO, MTS, and DPD systems in which production capability is considered. Furthermore, they analyzed the optimal point of postponement in a multistage queuing system. The DPD issue in manufacturing systems is studied by Jewkes and Alfa [2] in which they decided on where to locate the point of differentiation in a manufacturing system, and also what size of semifinished products inventory storage should be considered. In addition, they presented a model to realize how the degree of DPD affects the trade-off between customer order completion postponement and inventory risks, when both stages of production have nonnegligible time and the production capacity is limited. Also from a different point of view, the concept of order decoupling zone is introduced as an alternate to the DP concept by Wikner and Rudberg [14].

Recently, Ahmadi and Teimouri [15] studied the problem of where to locate the OPP in an auto export supply chain by using dynamic programming. Furthermore, a notable literature review in positioning DPs and studying the positioning of multiple DPs in a supply network can be seen in Sun et al. [16], but their positioning model did not make any decisions about the optimal semifinished buffer size and optimal fraction of processing time fulfilled by the upstream of DP. Jeong [17] developed a dynamic model to simultaneously determine the optimal position of the decoupling point and production–inventory plan in a supply chain. Also, many applications and methods for determining the OPP are surveyed in Olhager [18, 19], Yang and Burns [20], Yang et al. [21], Rudberg and Wikner [22], and Mikkola and Larsen [23].

The presented model tries to find equilibrium customer service levels with inventory costs, such as developed models in the literature. However, presented model differs from the studied articles in several ways. First, a two-stage MTS/MTO production model is used for each product type in a multiproduct, multiechelon supply chain. Second, the considered model gives the optimal transportation mode, optimal semifinished products warehouse capacity, and optimal fraction of processing time fulfilled by the supplier of each product type in an integrated model.

The supply chain which is considered as a basic model in this paper is composed of two production stages. In the first production stage, each product type's supplier supplies semifinished products on an MTS policy for a producer in the second production stage. The second stage producer will customize the products based on an MTO policy. The semifinished products will be completed as a result of specific customer orders. The supposed model obtains the optimal vehicles for the transportation of the completed products to each demand point.

In order to balance the costs of customer order fulfillment delay and inventory costs, each product type producer tries to find the optimal fraction of processing performed by the supplier and its optimal semifinished products buffer storage. The remainder of this paper is organized as follows. The problem description and formulation are reviewed in Sections 2 and 3. Also, the queuing aspect and performance evaluation indexes are studied in Section 3. Besides, the described model is studied with an additional warehouse capacity constraint in Section 4. Section 5 is dedicated to a numerical example. And finally, the study is concluded in Section 6.

2 Problem description and list of symbol

The following notations are used for the mathematical formulation of considered model.

Sets and indices:

i Products type index $i=1, 2,.., L$

m_i Semifinished products buffer storage capacity for product type i index m_i=1, 2,…, S_i
j Vehicles type index j=1, 2,.., J

Decision variables:

θ_i Percentage of completion for product type i in first production stage
S_i Optimal storage capacity of type i semifinished products
x_{ij} 1 if vehicle j is dedicated to logistic process of product type i, otherwise is 0.

Parameters:

$V(\theta_i)$ The value per unit of semifinished products (dollar/unit)
τ_i Constant fraction of the MTO processing rate for product type i
μ_i Production rate for product type i per each unit
C_{Ui} The cost of disposing an unsuitable item of type i (dollar/unit)
C_{Hi} The holding cost for semifinished products of type i for unit time (dollar/unit)
C_{Wi} The cost of customer order fulfillment delay for each unit of time for product type i (dollar/unit)
C_{Ci} The cost of establishing type i semifinished products storage capacity for each unit of time (dollar/unit)
c_{ij} Transportation cost of finished product type i by vehicle j for each unit of time (dollar/unit)
t_{ij} Transporting time for product type i with vehicle j
Cap_{ij} Capacity of vehicle j for product type i

Expected performance measures:

$E(N_i)$ The expected number of ith type semifinished products in the system
$E(W_i)$ The expected customer order completion delay for product type i—the time from when a customer order enters the system until its product is completed
$E(U_i)$ The expected number of ith type unsuitable products produced per unit time

A multiproduct multiechelon production supply chain is considered. In this system, it is assumed that the demands arrive according to a Poisson process with rate λ. Each customer orders one unit of ith product type with a probability of q_i where $\sum_{i=1}^{L} q_i = 1$ and $\lambda_i=\lambda q_i$, i=1, 2,…, L. The production times of workstations for all product types are assumed to be exponentially distributed with rates μ_i, i=1, 2,…, L where $\sum_{i=1}^{L} \mu_i = \mu$. These assumptions about arrival and service time's distributions return to the fact that the customer arrivals and system service times are memoryless. More specifically, the properties of random arrival and service times related to the future do not depend on any other information from further in the past. Therefore, the interval times between two consecutive arrivals follow an exponential distribution and demands arrival follow a Poisson process. These explanations are true for production time's exponential distributions. It is supposed that the supplier has an infinite source of raw materials and never faces shortage. The producer has to determine the optimal storage capacity of type i semifinished products (S_i, i=1, 2,…, L).

Each product type supplier produces a semifinished product [100% θ_i completed ($0<\theta_i<1$)] to be delivered to the producer. The producer then completes the remaining $1-\theta_i$ fraction according to a particular customer order. It should be noted that the supplier is not necessarily in a different organization from the producer; the "supplier" and "producer" may be two successive stages in a same organization. It is chosen to model θ_i as a continuous variable so that greater insights into the overall relationship between θ_i and the performance of the system can be gained. The assumption also facilitates the computational analysis. Therefore, results are presented as if the producer can implement any values of θ_i. If this is not the case, the model enables the managers to quickly identify the best choice of θ_i among a finite number of feasible alternatives. According to market characteristics studied by Jewkes and Alfa [2], there is a probability of $\phi_i(\theta_i)$ that a semifinished product is not suitable for customization and so $\phi_i(\theta_i)$ is monotonically increasing with θ_i which is a rational assumption. Figure 1 illustrates a diagram depicting proposed model.

It is assumed that there is some logistics time to supply items from producers to demand points. The logistics process is modeled by using queuing notation $M/D^{\mathrm{Cap}}{}_{ij}/\infty$ in a continuous time (as discussed by Purdue and Linton [24] and Kashyap et al. [25]), where M denotes the exponential arrivals of completed products to logistics process which is logical, owing to semifinished products completion time, D represents that each vehicle service time is deterministic, Cap_{ij} is the jth vehicle capacity for product type i which is deterministic, and the vehicles do not transport any product up to the time they become filled to capacity. It is also assumed that infinite vehicles are available to supply the order; this assumption is represented by ∞ in the queuing notation. These transport assumptions hold good for third party logistics which can be applied in practical situations.

3 Problem formulation

The entire explained system in Section 2 can be described by a Markov process with state (n_i, m_i), where n_i is the number of customers in the system waiting for each finished product type i and m_i is the number of type i semifinished products in its semifinished product storage [2]. Therefore,

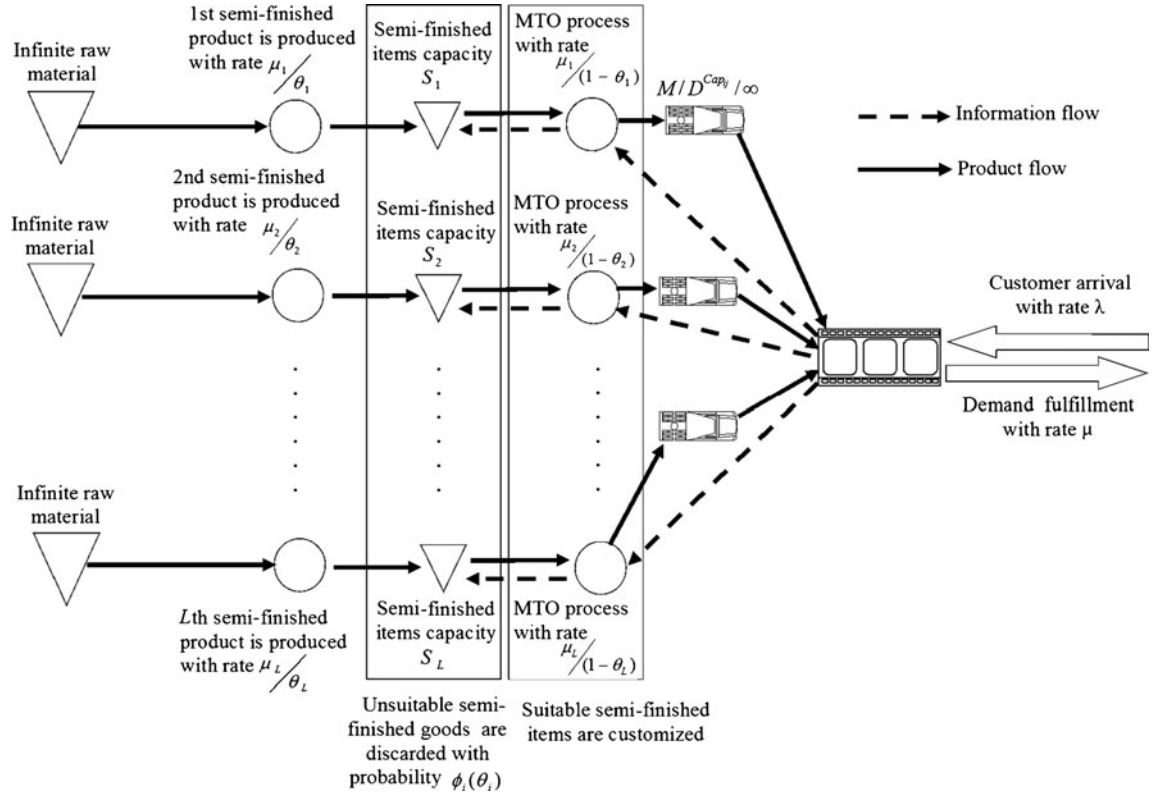


Fig. 1 The multiproduct hybrid MTO/MTS production supply chain system

the state space is denoted by $\Omega = \{n_i \geq 0, 0 \leq m_i \leq S_i\}$, which is depicted in Fig. 2 with transition rates.

In Fig. 2 (a, b) stands for $\frac{\mu(1-\phi)}{\theta}$ and $\frac{\mu}{1-\theta}$ for each product type, respectively. The associated balance equations for the

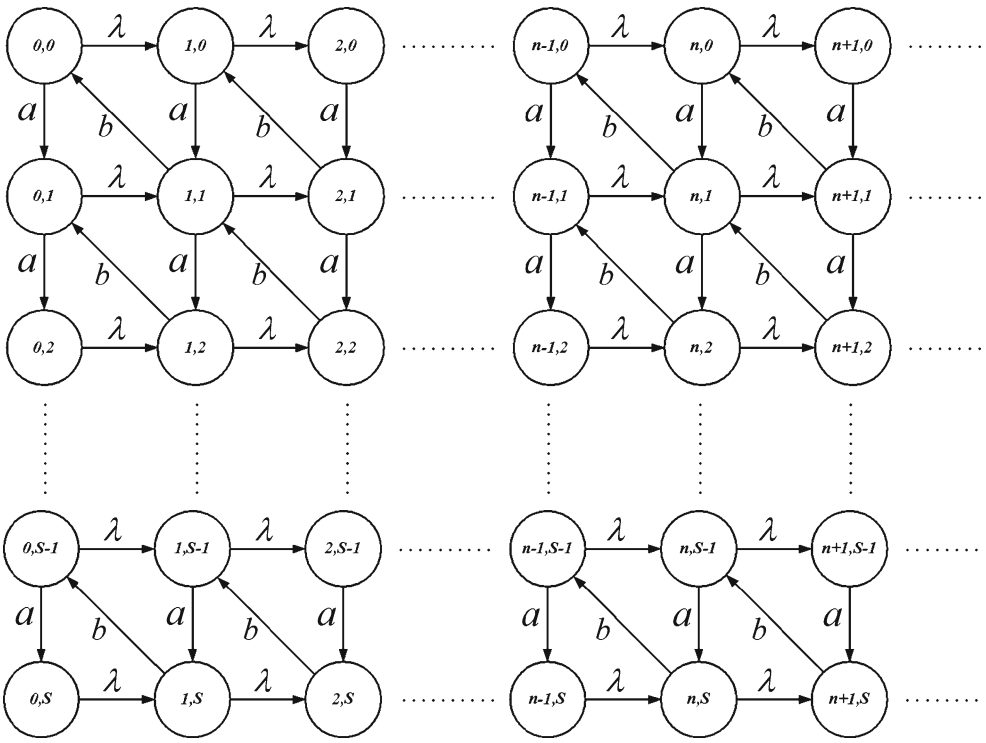


Fig. 2 State transition rates diagram

steady probabilities follow Eq. 1, 2, 3, 4, 5, and 6.

$$\left(\frac{\mu_i(1-\phi_i)}{\theta_i}+\lambda_i\right)P_i(n_i,m_i)=\frac{\mu_i}{1-\theta_i}P_i(n_i+1,m_i+1),\quad n_i=0,m_i=0 \tag{1}$$

$$\left(\frac{\mu_i(1-\phi_i)}{\theta_i}+\lambda_i\right)P_i(n_i,m_i)=\frac{\mu_i(1-\phi_i)}{\theta_i}P_i(n_i,m_i-1)+\frac{\mu_i}{1-\theta_i}P_i(n_i+1,m_i+1),\quad n_i=0,1\le m_i\le S_i-1 \tag{2}$$

$$\frac{\mu_i(1-\phi_i)}{\theta_i}P_i(n_i,m_i-1)=\lambda_i P_i(n_i,m_i),\qquad n_i=0,m_i=S_i \tag{3}$$

$$\left(\frac{\mu_i(1-\phi_i)}{\theta_i}+\lambda_i\right)P_i(n_i,m_i)=\lambda_i P_i(n_i-1,m_i)+\frac{\mu_i}{1-\theta_i}P_i(n_i+1,m_i+1),\quad 1\le n_i,m_i=0 \tag{4}$$

$$\left(\frac{\mu_i(1-\phi_i)}{\theta_i}+\lambda_i+\frac{\mu_i}{1-\theta_i}\right)P_i(n_i,m_i)=\frac{\mu_i(1-\phi_i)}{\theta_i}P_i(n_i,m_i-1)+\frac{\mu_i}{1-\theta_i}P_i(n_i+1,m_i+1)+\lambda_i P_i(n_i-1,m_i),\quad n_i=0,1\le m_i\le S_i-1 \tag{5}$$

$$\left(\lambda_i+\frac{\mu_i}{1-\theta_i}\right)P_i(n_i,m_i)=\frac{\mu_i(1-\phi_i)}{\theta_i}P_i(n_i,m_i-1)+\lambda_i P_i(n_i-1,m_i),\quad 1\le n_i,m_i=S_i \tag{6}$$

There exists the corresponding generator matrix Q_i written in block form (Eq. 7) for the product type i:

$$Q_i=\begin{bmatrix} G_i & A_i & & & \\ C_i & E_i & A_i & & \\ & C_i & E_i & A_i & \\ & & \ddots & \ddots & \ddots \end{bmatrix} \tag{7}$$

Appendix A shows block matrices where A_i, C_i, E_i, and G_i are block matrices with the dimension of $(S_i+1)\times(S_i+1)$. It is notable that A_i giving the rate at which the number of customer orders in the system increases by one, E_i giving the rate at which the number of customer orders in the system either stays at the same level, and C_i giving the rate at which the number of customer orders in the system decreases by one. G_i is the matrix rate at which the customer orders in the system move from zero to one.

Let $F_i=A_i+E_i+C_i$ be a generator matrix with its associated stationary distribution $P_i=[P_{i0},P_{i1},\ldots,P_{iS_i}]$ given as a solution to $P_iF_i=0$, $P_i\mathbf{1}=1$.

$$F_i=\begin{bmatrix} F_{i_{0,0}} & F_{i_{0,1}} & & & \\ F_{i_{1,0}} & F_{i_{1,1}} & F_{i_{1,2}} & & \\ \ddots & \ddots & \ddots & & \\ & F_{i_{S_i-1,S_i-2}} & F_{i_{S_i-1,S_i-1}} & F_{i_{S_i-1,S_i}} & \\ & & F_{i_{S_i,S_i-1}} & F_{i_{S_i,S_i}} & \end{bmatrix} \tag{8}$$

Appendix B illustrates block matrices where $F_{i_{m,m+1}}$, $F_{i_{m,m-1}}$, and $F_{i_{m,m}}$ are $(S_i+1)\times(S_i+1)$. As it is discussed in Neuts [26], the explained Markov chain is stable if $P_iC_i\mathbf{1}>P_iA_i\mathbf{1}$. In order to have a stable system, the producer requires having a service rate that exceeds the arrival rate of customers. In addition, the supply rate of suitable semifinished products to the producer must be more than the customer demands rate.

3.1 Steady-state analysis

The behavior of this supply chain system is studied in a steady state. Let $\Pi_i=[\Pi_{i\mathbf{0}},\Pi_{i\mathbf{1}},\Pi_{i\mathbf{2}},\ldots]$ be the stationary probabilities associated with the Markov chain for each product type so that $\Pi_iQ_i=0$ and $\Pi_i\mathbf{1}=1(i=1,2,\ldots,L)$. Due to the matrix geometric theorem [26], equation $\Pi_{i,n+1}=\Pi_{i,n}R_i,\quad n\ge 0$ must be satisfied where R_i is the minimal nonnegative solution to the matrix quadratic equation $A_i+R_iE_i+R_i^2C_i=0$.

It is noteworthy that matrix R_i can be computed very easily using some well-known methods according to Bolch et al. [27]. A simple way to compute R_i is the iterative approach given as $R_i(n+1)=-\left(A_i+R_i(n)^2C_i\right)E_i^{-1}$ until $|R_i(n+1)-R_i(n)|_{nj}<\varepsilon$, with $R_i(0)=0$. The boundary vector Π_{i0} is obtained from $\Pi_{i0}(G_i+R_iC_i)=0$.

3.2 Performance evaluation indexes

Here, the important performance evaluation indexes of the system can be obtained as described below. Let $E[O_i]$ be the mean number of customers' orders for product type i in the system, including the one being served; $E[W_i]$ be the mean customer order completion delay for product type i; $E[N_i]$ be the mean number of semifinished products in the system for product type i; and $E[U_i]$ be the expected number of unsuitable semifinished products disposed per unit time for product type i, then

$$E[O_i]=\Pi_{i1}(I-R_i)^{-2}\mathbf{1}$$

$E[W_i] = \frac{E(O_i)}{\lambda_i}$ (by applying Little's Law), $E[N_i] = \Pi_{i0}(I - R_i)^{-1} y_i$, where $y_i = [0,1,2,\ldots,S_i]^T$, and $E(U_i) = \frac{(1-\Pr(m_i=S_i))\phi_i\mu_i}{\theta_i}$, where m_i denotes the number of semifinished products storage for each product type.

3.3 Mathematical model

The objective function includes the following costs:

1. Disposing of semifinished products that are not appropriate for customizing the customer orders ($C_{U_i} V(\theta_i) E(U_i)$),
2. Holding semifinished products in buffer storage ($C_{H_i} V(\theta_i) E(N_i)$),
3. Providing storage capacity for the semifinished products ($C_{C_i} S_i$).
4. Customer order fulfillment delay ($\sum_{i=1}^{L}\sum_{j=1}^{J} x_{ij} C_{W_i} \left(\mathrm{Cap}_{ij} \cdot E(W_i) + t_{ij}\right)$), and
5. Transportation cost ($\sum_{i=1}^{L}\sum_{j=1}^{J} c_{ij} \cdot \mathrm{Cap}_{ij} \cdot x_{ij}$).

The mathematical formulation of the model is as follows:

$$\min_{S_i,\theta_i,x_{ij}} \mathrm{TC}\left(S_i,\theta_i,x_{ij}\right) = C_{U_i} V(\theta_i) E(U_i) + C_{H_i} V(\theta_i) E(N_i) + C_{C_i} S_i + \sum_{i=1}^{L}\sum_{j=1}^{J} x_{ij} C_{W_i} \left(\mathrm{Cap}_{ij} \cdot E(W_i) + t_{ij}\right) + \sum_{i=1}^{L}\sum_{j=1}^{J} c_{ij} \cdot \mathrm{Cap}_{ij} \cdot x_{ij} \tag{9}$$

subject to:

$$\sum_{j=1}^{J} x_{ij} = 1 \qquad \forall i \tag{10}$$

$$\tau_i \frac{\mu_i}{(1-\theta_i)} \le \frac{1}{E(W_i)} + \frac{\mathrm{Cap}_{ij}}{t_{ij}} \qquad \forall i \tag{11}$$

$$0 < \theta_i < 1.0 \quad \forall i \tag{12}$$

$$S_i = 1,2,\ldots \quad \forall i \tag{13}$$

$$x_{ij} \in \{0,1\} \qquad \forall i,j \tag{14}$$

The model (Eq. 9) minimizes the total expected cost including the cost of semifinished products that are not consistent with customer's order, expected semifinished products holding cost, the cost of establishing storage capacity for semifinished products, expected cost of delay in customer order completion (which include time of customization and logistics), and transportation costs based on the vehicle type selected. Constraint in Eq. 10 assures that logistics process for each product type accomplishes by exactly one vehicle. Constraint in Eq. 11 represents that a constant value (τ_i) of the MTO processing rate for product type i ($\frac{\mu_i}{(1-\theta_i)}$) must be at its most less than the total customer order completion rate which contains customization and logistics process (service level constraint). Constraints in Eqs. 12, 13, and 14 represent the ranges of the model variables.

In order to solve proposed mathematical model, a direct search heuristic method is used. The values of S_i and θ_i must vary in their allowable variation ranges to find their near optimal values. These various values of storage capacity and completion percentage enable us to calculate system performance measures which are used in mathematical model. The outputs of the represented model are the optimal fractions of the process fulfilled by the supplier for each product type, their optimal sem-finished products buffer storage capacity, and the optimal transportation mode for each product type.

4 Studying model under the warehouse capacity constraint

This section studies a more realistic constraint that can be added to the proposed model (see Fig. 3). According to warehouses physical structure, it is not possible to establish every calculated optimal storage capacity for each product type. This is a logical assumption in operational problems. In this section, specific capacity of K is considered for semifinished product warehouse.

Due to separate calculations of optimal storage capacity for each product type, the storage space constraint cannot be applied in the optimization model. Therefore, if the summation of semifinished product storage related to obtained optimal solution for all types of products satisfies the warehouse capacity constraint, the obtained solutions can be considered as optimal storage capacities. However, if the warehouse capacity constraint has not been satisfied, the developed heuristic solution procedure can be used as follows.

Algorithm

Step 1 Find the optimal values S_i^* and θ_i^* for each product type.

Step 2 Calculate $\sum_{i=1}^{L} S_i^*$ (cumulative storage value for all product types). If $\sum_{i=1}^{L} S_i^*$ is smaller than the predefined capacity constraint for central warehouse (K), solutions in step 1 are acceptable: stop. Otherwise, use step 3.

Step 3 Find $\mathrm{TC}_i\left(S_i^* - 1, \theta_i\left(S_i^* - 1\right)\right) - \mathrm{TC}_i(S_i^*, \theta_i^*)$ for each product type. Set $S_i^* - 1 \rightarrow S_i^*$ (if $S_i^* - 1$ is stable) and $\theta_i(S_i^* - 1) \rightarrow \theta_i^*$ for product type with the lowest $\mathrm{TC}_i\left(S_i^* - 1, \theta_i\left(S_i^* - 1\right)\right) - \mathrm{TC}_i(S_i^*, \theta_i^*)$.

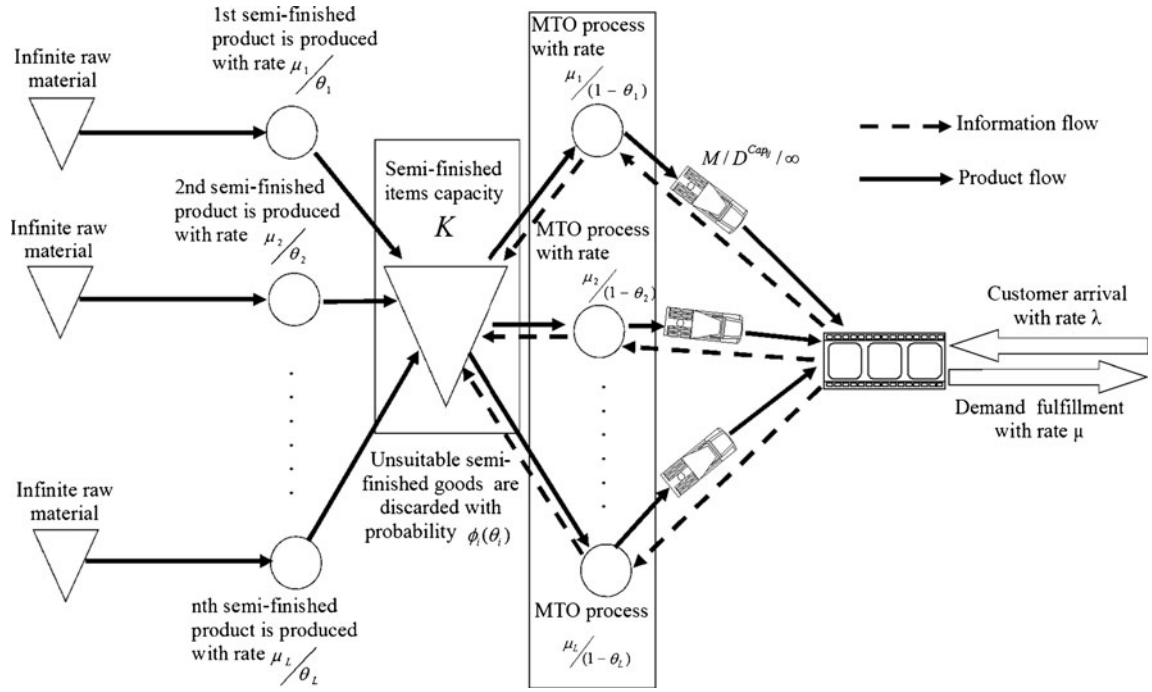


Fig. 3 The multiproduct hybrid MTO/MTS production supply chain with the capacitated warehouse

Step 4 If $\sum_{i=1}^{L} S_i^* \le K$, solutions obtained in step 3 are acceptable: stop. Otherwise, use step 5.

Step 5 Go to step 3.

The proposed algorithm is represented schematically in Fig. 4. Although the developed algorithm is so time-consuming due to the enumeration technique used in its steps, it computes a nearly optimal solution with minimum benefit loss.

5 Numerical example

In this section a numerical example is used to show the relation between $\mathrm{TC}(S_i, \theta_i(S_i))$ and variables $\theta_i^*(S_i)$ and S_i, also system analysis is done for a variety of parameters. A production supply chain network containing three product types with three suppliers, three producers, a capacitated warehouse with the capacity of $K=7$, and three types of transportation vehicles is considered. The following parameter values are considered which are changeless for the

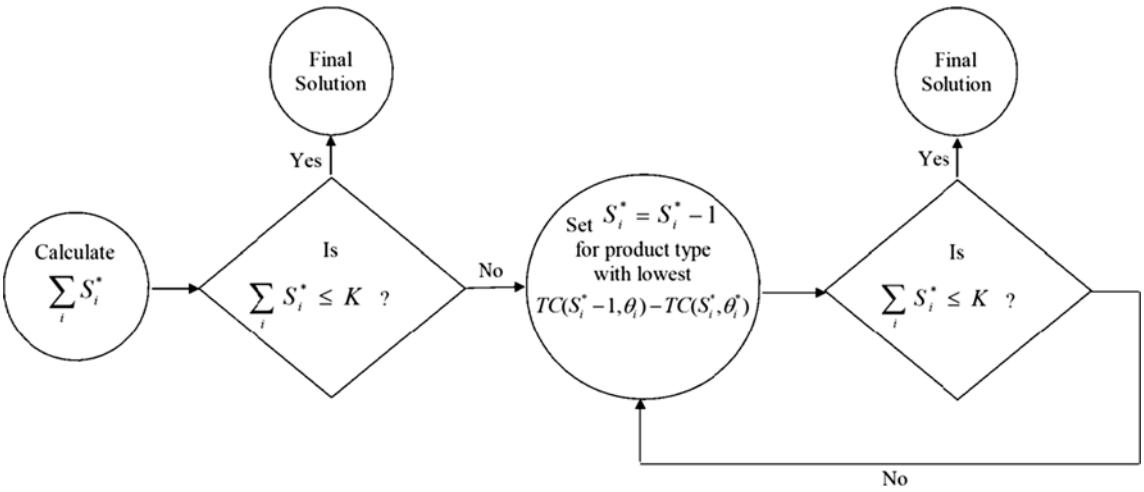


Fig. 4 Heuristic solution procedure

various product types: τ_i=0.05, C_W=1.2, C_H=0.1, and each semifinished product value $V(\theta_i)$ equals to θ_i as assumed by Jewkes and Alfa [2]. Other necessary information about each product type and transportation vehicles characteristics id provided in Table 1 (λ_i is calculated by λq_i for each product type).

As shown in Table 2, the system is not stable for any S_i= 1 due to the supplier disability to provide sufficient suitable semifinished products to satisfy the producers' demands. The optimal values of storage capacities and the process postponement for each product type are shown in bold in Table 2. As it is seen, the semifinished product completed percentage is an ascending function of capacity growth. It is notable that when there is a lower capacity, the semifinished products are highly affected by demand fluctuations and the manufacturer prefers to bear the cost of completion delay for a more customization right and preventing of disposing semifinished products. But as it is seen, by increasing the storage capacity, the affect of demand fluctuations in cost function decreases and the manufacturer prefers to increase the completed percentage in first echelon in order to reduce the product completion delay.

Also there exists a trend in total cost function which is affected by cost parameters. For product type 1 and 3, a reduction in total cost function can be seen which can be explained by the increase of product completed part due to storage growth. This increase in product completed part reduces the completion delay cost which is more than the growth of other four cost parameters. But there exists an increasing trend for total cost function after S_i=3 for product type 1 and 3 and also for all capacities of product type 2, which is reasonable due to structure of cost parameters. It is obvious that holding semifinished products and providing storage capacity are increasing functions of capacity growth which are more effective than other cost parameters and finally increase the total cost function by growing the semifinished product storage. In order to better understand the affect of completion percentage on total cost, the variation of total cost function versus completion percentage of semifinished product type 1 is shown in Fig. 5, it is notable that zero total costs are related to infeasible points.

Table 2 Results of numerical example

Product type	S_i	Optimal vehicle	$\theta_i^*(S_i)$	Total cost
1	1	3	Nonstable	Nonstable
	2		0.26	15.0867
	3		0.29	**15.0300**
	4		0.29	15.2436
	5		0.30	15.6314
	–		–	–
	30		0.30	25.6630
	40		0.30	29.6633
	50		0.30	33.6634
2	1	3	Nonstable	Nonstable
	2		**0.28**	**12.3716**
	3		0.30	12.8561
	4		0.30	13.5357
	–		–	–
	30		0.31	31.7625
	40		0.31	38.7626
	50		0.31	45.7626
3	1	3	Nonstable	Nonstable
	2		0.28	14.0539
	3		**0.31**	**13.9608**
	4		0.32	14.2431
	5		0.33	14.6173
	–		–	–
	30		0.33	24.6454
	40		0.33	28.6457
	50		0.33	32.6458

In this example an optimal transportation vehicle is selected for each product type. It is worth noting that the logistic process does not follow an assignment model, and each vehicle type can be used for more than one product type. The derived results of change trend in Fig. 5 are in accordance with Jewkes and Alfa [2].

Due to warehouse capacity, the satisfaction condition $\sum_{i=1}^{L} S_i^* \le K$ must be checked and if the storage capacity

Table 1 Parameters' data for numerical example

Product type	λ_i	μ_i	ϕ_i	C_{c_i}	$C_{U\bar{i}}$	Transport time by vehicle j			Transport cost by vehicle j			Transport capacity by vehicle j		
						1	2	3	1	2	3	1	2	3
1	0.7	1	$0.9\theta_1$	0.4	1	10	8	5	0.25	0.26	0.3	5	4	3
2	0.6	1	$0.7\theta_2$	0.7	0.8	10	8	5	0.30	0.32	0.38	3	2	2
3	0.9	1.2	$0.75\theta_3$	0.4	0.7	10	8	5	0.19	0.21	0.28	4	3	3

The computational results are based on the MATLAB 7.1 implementation where the total cost is computed for $0.01 \le \theta_i \le 0.99$ in increments of 0.01 where S_i varies from 1 to 50

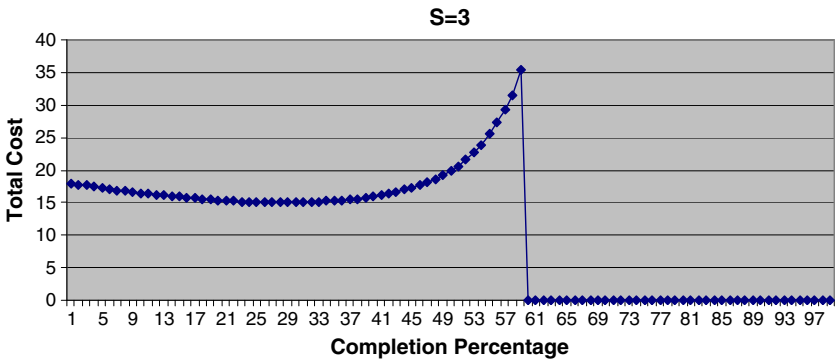


Fig. 5 $TC(S_i, \theta_i(S_i))$ versus $\theta_i^*(S_i)$ for product type 1

constraint does not hold, the developed heuristic solution must be run:

Step 2 $\sum_{i=1}^{L} S_i^* = 3 + 2 + 3 > K = 7$

Step 3 $\left.\begin{array}{l} \mathrm{TC}_1(2,0.26) - TC_1(3,0.29) = 15.08 - 15.03 = 0.05 \\ \mathrm{TC}_2(1,\theta_2(1)) = \text{nonstable} \\ \mathrm{TC}_3(2,0.28) - \mathrm{TC}_1(3,0.31) = 14.05 - 13.96 = 0.09 \end{array}\right\} \Rightarrow S_1^* = 2, \theta_1^* = 0.26$

Step 4 $\sum_{i=1}^{L} S_i^* = 2 + 2 + 3 \leq K = 7$

Now the near optimal solutions with minimum benefit loss are obtained. The storage capacity for first, second, and third type products equals to 2, 2, and 3, respectively. Moreover, the OPP must stand after the 0.26, 0.28, and 0.31 of product completion in each product's supply chain and the transportation vehicle for all products still remains the third one due to the explained reasons. In Section 5 the system manner under different parameter variations must be analyzed. The logical manner of surveyed system characteristics can be considered as a validation for the proposed model and the performed computations.

5.1 Affect of demand λ fluctuations on total cost function and OPP location

According to queuing theory fundamentals, the customer arrival rate must be lower than systems' service rate due to establish system stability; otherwise, the queue length goes infinite. In this example the service rate for product type 1 is equal to 1, so the affect of varying customer arrival rate on the total system cost is studied by the values between 0.2 and 0.9 which increments by 0.1. Changes in total cost function for various values of customer arrival rate are shown in Fig. 6. It is notable that zero total costs are related to infeasible points.

Higher arrival rate enhances the system busy time, and the queue length and customer order fulfillment time are increased, consequently. It is obvious that the total cost of system will be augmented by increasing system busy time, due to the mentioned explanations and the fact that there is no unemployment cost for the system.

In addition to the affect of increasing λ_i on positioning, OPP is remarkable. As it is shown in Fig. 7, increasing λ_i has an increasing effect on optimal value of OPP. The derived results of change trend in Fig. 6 are in accordance with Jewkes and Alfa [2].

Moreover, the place of OPP differentiates with different values of demand rate. It is obvious that the expected number of customers will increase by growing demand, therefore the queue length will raise and customers must wait a longer time to get service. This fact enforces the manufacturer to complete a more percentage of products in the first echelon and satisfies the demand with less delay. This increase in OPP reduces the queue length and completion delay which follows the reduction in the total cost function.

5.2 Affect of production rate μ fluctuations on system performance measures

Increasing production rate has a specific effect on each system performance measures such as increasing service rate and customer satisfaction, but the cost of increasing production rate is a preventive factor of enjoying these advantages. In this section the affect of various production rates on some system performance measures is studied and shown schematically in Fig. 8 (μ stands for production rate).

In Fig. 8a the expected order completion delay is reduced by increasing production rate which is justifiable by queue length. The higher production rate leads to a higher demand satisfaction rate, therefore the queue length would be decreased and this is equal to less completion delay due to Little's laws. The affect of production rate on expected number of unsuitable products is shown in Fig. 8b. The production rate μ is used as a linear coefficient in calculating the expected number of unsuitable products, and without any conceptual explanations, it is expectable to have an increasing manner of $E(U_i)$ by growing μ.

The expected number of semifinished products in the system decreases versus production growth, because the higher rate of production satisfies the customer demands with a higher service rate and a lower value of semifinished

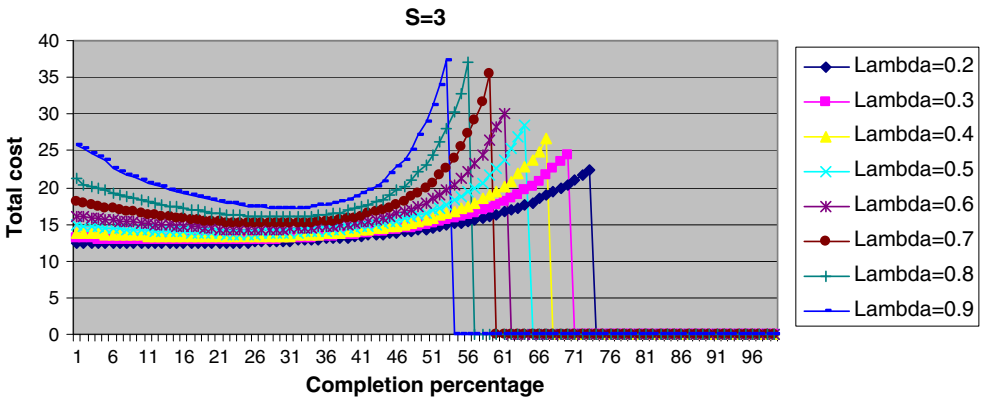


Fig. 6 $TC(S_i, \theta_i(S_i))$ versus $\theta_i^*(S_i)$ and λ_i for product type 1

products would be remained in the system, consequently. This reduction of semifinished products against production rate is depicted in Fig. 8c.

5.3 Affect of various capacity storages on system performance measures

Increasing of expected number of semifinished products is the first thing which is expected due to increasing of storage capacity. But growing the storage capacity values will affect the other system performances which are depicted in Fig. 9 schematically.

As it is discussed, growing storage capacity enhances the expected number of semifinished products which is shown in Fig. 9a. This growing trend is approximately linear which is expected from the initial numerical example results where the minimum costs are related to lower values of capacity storage.

Figure 9b shows the expected number of unsuitable products growth. This trend can be interpreted by referring to the formula of calculating unsuitable products. The probability of storage not being full is used, and it is obvious that growing storage capacity will reduce the probability of a full storage and enhance the probability of having an empty storage. The probability of having empty storage is multiplied to the numerator of unsuitable products calculating formula, so the growth of storage capacity will enhance the expected number of semifinished products which must be discarded.

Figure 9c is dedicated to showing the variations of expected order completion delay for storage capacity. Growing storage capacity will enhance the expected number of semifinished products in the system, and the demands will satisfied with higher rate which reduces the queue length. As it was discussed, lower queue length will logically reduce the completion delay. The derived results of change trend in Fig. 9 are in accordance with Jewkes and Alfa [2].

6 Conclusion

OPP is the boundary between MTO and MTS policies. In this article an optimization model was presented to determine OPP in a multiproduct multiechelon supply chain. The affects of product customization postponement on customer order completion delay and inventory risks were discussed. In order to evaluate performance measures, a simple queuing model and an explicitly matrix geometric method was applied.

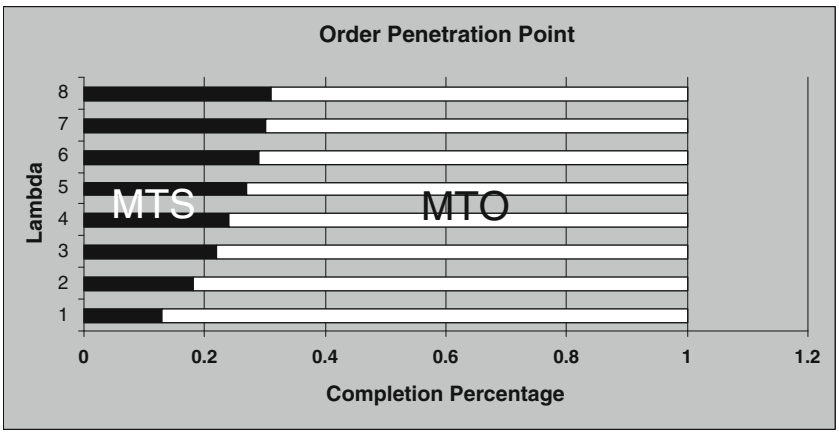
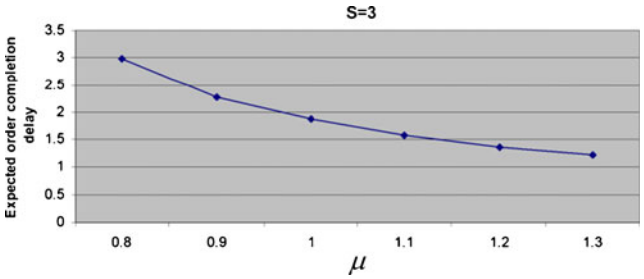
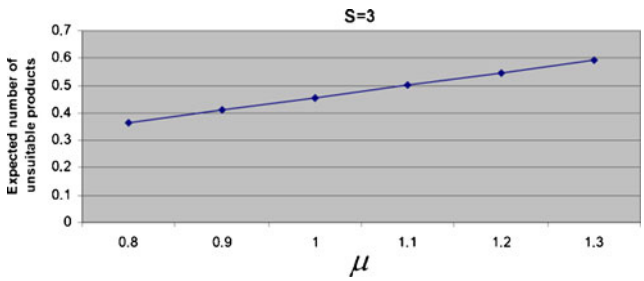


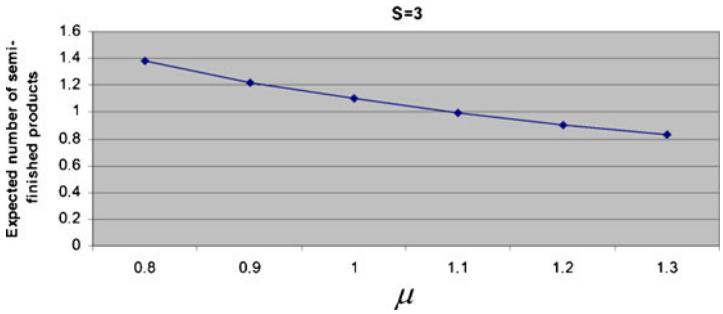
Fig. 7 OPP versus λ_i for product type 1



A. Expected order completion delay versus μ



B. Expected number of unsuitable products versus μ

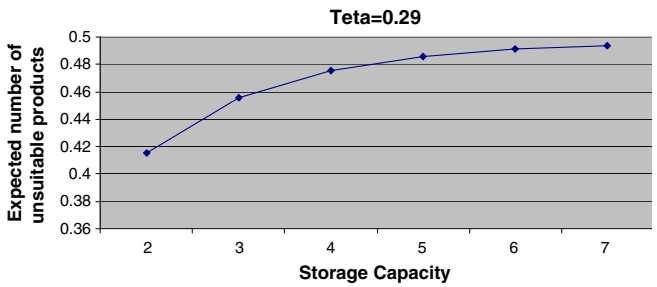


C. Expected number of semi-finished products versus μ

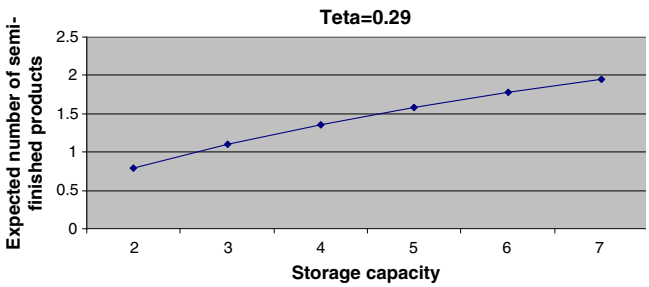
Fig. 8 **a**–**c** System performance measures versus μ

The proposed model aims to obtain the optimal OPP in a supply chain, optimal level of buffer storage capacity, as well as the best finished products transporting vehicle for each product type. The transportation is modeled as a logistic process where each vehicle has a constant capacity and a deterministic delivery time. In addition, the problem with a real warehouse capacity constraint is considered as a development of the main model of the article.

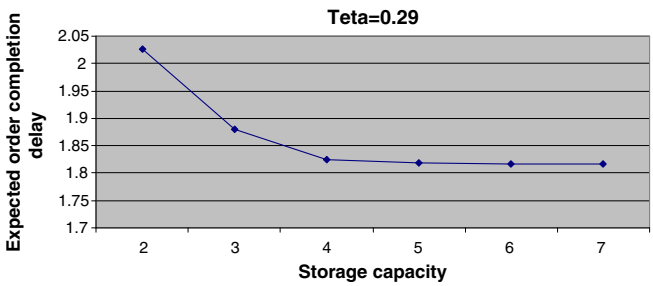
The numerical example and the sensitivity analysis explain the system manner under various chain parameters. It



B. Expected number of unsuitable products versus storage capacity



A. Expected number of semi-finished products versus storage capacity



C. Expected order completion delay versus storage capacity

Fig. 9 **a**–**c** System performance measures versus storage capacity

is shown that the total cost function increases by the increasing demand arrival rate, and it is concave in increasing completed percentage of semifinished products. Furthermore, it is shown that the semifinished product completed percentage is an ascending function of capacity growth by the considered assumptions of considered model. The observation of arrival demand fluctuations showed the increase of optimal OPP due to increasing values of lambda. Moreover, the affect of production rate μ and capacity storage fluctuations show that the expected number of semifinished products is decreasing in μ but increasing in storage capacity, the expected number of unsuitable products is increasing in both μ and storage capacity and the expected order completion delay is decreasing in both μ and storage capacity.

Manufacturers must consider the storage capacity, disposal, and order completion delay costs as the important decision-making parameters. As it is obvious the increasing OPP would increase the storage holding and disposal costs and decrease the completion delay cost. Also, decreasing OPP has an opposite effect on holding, disposal, and completion delay costs. Manufacturers must locate the OPP where the related costs get balanced, and the summation of all considered costs must be minimized. The rates of service and customer arrivals are effective factors which must be considered on choosing OPP, too. Applying the capacity constraint in customers queue, relaxing the assumptions of exponentially distributed arrival and service times, and considering the impatient customers in arrival demands can be as future research possibilities.

Acknowledgment The authors are thankful for constructive comments of the reviewers and the editor that certainly improved the presentation of the paper.

Appendix A

$$G_i = \begin{bmatrix} G_{i_{0,0}} & G_{i_{0,1}} & & \\ & G_{i_{1,1}} & G_{i_{1,2}} & \\ & \ddots & \ddots & \\ & & G_{i_{S_i-1,S_i-1}} & G_{i_{S_i-1,S_i}} \\ & & & G_{i_{S_i,S_i}} \end{bmatrix}_{(S_i+1)\times(S_i+1)}$$

$$G_{i_{m,m}} = \begin{cases} -\left(\lambda_i + \frac{\mu_i(1-\phi_i)}{\theta_i}\right) & 1 \le i \le L, \quad 0 \le m \le S_i - 1 \\ -\lambda_i & 1 \le i \le L, \quad m = S_i \end{cases}$$

$$G_{i_{m,m+1}} = \frac{\mu_i(1-\phi_i)}{\theta_i} \qquad 1 \le i \le L, \quad 0 \le m \le S_i - 1 \tag{A.1}$$

$$E_i = \begin{bmatrix} E_{i_{0,0}} & E_{i_{0,1}} & & \\ & E_{i_{1,1}} & E_{i_{1,2}} & \\ & \ddots & \ddots & \\ & & E_{i_{S_i-1,S_i-1}} & E_{i_{S_i-1,S_i}} \\ & & & E_{i_{S_i,S_i}} \end{bmatrix}_{(S_i+1)\times(S_i+1)}$$

$$E_{i_{m,m}} = \begin{cases} -\left(\lambda_i + \frac{\mu_i(1-\phi_i)}{\theta_i}\right) & 1 \le i \le L, \quad m = 0 \\ -\left(\lambda_i + \frac{\mu_i(1-\phi_i)}{\theta_i} + \frac{\mu_i}{1-\theta_i}\right) & 1 \le i \le L, \quad 1 \le m \le S_i - 1 \\ -\left(\lambda_i + \frac{\mu_i}{1-\theta_i}\right) & 1 \le i \le L, \quad m = S_i \end{cases}$$

$$E_{i_{m,m+1}} = \frac{\mu_i(1-\phi_i)}{\theta_i} \qquad 1 \le i \le L, \quad 0 \le m \le S_i - 1 \tag{A.2}$$

$$C_i = \begin{bmatrix} 0 & 0 \\ I\frac{\mu_i}{1-\theta_i} & 0 \end{bmatrix}_{(S_i+1)\times(S_i+1)} \tag{A.3}$$

$$A_i = [I\lambda_i]_{(S_i+1)\times(S_i+1)} \tag{A.4}$$

Appendix B

$$F_{i_{m,m}} = \begin{cases} -\left(\frac{\mu_i(1-\phi_i)}{\theta_i}\right) & 1 \le i \le L, \quad m = 0 \\ -\left(\frac{\mu_i(1-\phi_i)}{\theta_i} + \frac{\mu_i}{1-\theta_i}\right) & 1 \le i \le L, \quad 1 \le m \le S_i - 1 \\ -\left(\frac{\mu_i}{1-\theta_i}\right) & 1 \le i \le L, \quad m = S_i \end{cases} \tag{B.1}$$

$$F_{i_{m,m+1}} = \frac{\mu_i(1-\phi_i)}{\theta_i} \qquad 1 \le i \le L, \quad 0 \le m \le S_i - 1 \tag{B.2}$$

$$F_{i_{m,m-1}} = \frac{\mu_i}{1-\theta_i} \qquad 1 \le i \le L, \quad 1 \le m \le S_i \tag{B.3}$$

References

1. Chopra S, Meindl P (2003) Supply chain management: strategy, planning, and operations. Prentice Hall, 2nd edn, 592 pages, ISBN-13: 978-0131010284
2. Jewkes EM, Alfa AS (2009) A queuing model of delayed product differentiation. Eur J Oper Res 199:734–743
3. Hajfathaliha A, Teimoury E, khondabi IG, Fathi M (2011) Using queuing approach for locating the order penetration point in a two-echelon supply chain with customer loss. Int J Bus Res Manag 6(1) ISSN 1833-3850;E-ISSN 1833-8119
4. Teimoury E, Modarres M, Kazeruni Monfared A, Fathi M (2011) Price, delivery time, and capacity decisions in an M/M/1 make-to-order/service system with segmented market. Int J Adv Manuf Technol. doi:10.1007/s00170-011-3261-2
5. Sharman G (1984) The rediscovery of logistics. Harv Bus Rev 62 (5):71–79
6. Adan IJBF, Van der Wal J (1998) Combining make to order and make to stock. OR-Spektrum 20(2):73–81
7. Arreola-Risa A, DeCroix GA (1998) Make-to-order versus make-to-order in a production-inventory system with general production times. IIE Trans 30(8):705–713
8. Günalay Y (2010) Efficient management of production-inventory system in a multi-item manufacturing facility: MTS vs MTO. Int J Adv Manuf Technol 54:1179–1186. doi:10.1007/s00170-010-2984
9. Rajagopalan S (2002) Make-to-order or make-to-stock: model and application. Manag Sci 48(2):241–256
10. Perona M, Saccani N, Zanoni S (2009) Combining make-to-order and make-to-stock inventory policies: an empirical application to a manufacturing SME. Prod Plan Control 20(7):559–575. doi:10.1080/09537280903034271
11. Quante R, Meyr H, Fleischmann M (2009) Revenue management and demand fulfillment: matching applications, models, and software. OR Spectr 31(1):31–62. doi:10.1007/s00291-008-0125-8
12. Aviv Y, Federgruen A (2001) Design for postponement: a comprehensive characterization of its benefits under unknown demand distributions. Oper Res 49(4):578–598
13. Gupta D, Benjaafar S (2004) Make-to-order, make-to-stock, or delay product differentiation? A common framework for modeling and analysis. IIE Trans 36:529–546
14. Wikner J, Rudberg M (2005) Introducing a customer order decoupling zone in logistics decision making. Int J Logist Res Appl 8 (3):211–224
15. Ahmadi M, Teimouri E (2008) Determining the order penetration point in auto export supply chain by the use of dynamic programming. J Appl Sci 8(18):3214–3220
16. Sun XY, Jib P, Sun LY, Wang YL (2008) Positioning multiple decoupling points in a supply network. Int J Prod Econ 113:943–956
17. Jeong IJ (2011) A dynamic model for the optimization of decoupling point and production planning in a supply chain. Int J Prod Econ 131(2):561–567
18. Olhager J (2003) Strategic positioning of the order penetration point. Int J Prod Econ 85:319–329
19. Olhager J (2010) The role of the customer order decoupling point in production and supply chain management. Comput Ind 61 (9):863–868
20. Yang B, Burns ND (2003) Implications of postponement for the supply chain. Int J Prod Res 41(9):2075–2090
21. Yang B, Burns ND, Backhouse CJ (2004) Postponement: a review and an integrated framework. Int J Oper Prod Manag 24(5):468–487
22. Rudberg M, Wikner J (2004) Mass customization in terms of the customer order decoupling point. Prod Plan Control 15(4):445–458
23. Mikkola JH, Larsen TS (2004) Supply chain integration: implications for mass customization, modularization and postponement strategies. Prod Plan Control 15(4):352–361
24. Purdue P, Linton D (1981) An infinite-server queue subject to an extraneous phase process and related models. J Appl Probab 18:236–244
25. Kashyap BRK, Liu L, Templeton JGC (1990) On the $GI^X/G/\infty$ system. J Appl Probab 27:671–683
26. Neuts MF (1981) Matrix-geometric solutions in stochastic models: an algorithmic approach. Johns Hopkins University Press, Baltimore
27. Bolch G, Greiner S, De Meer H, Trivedi KS (2006) Queueing networks and Markov chains: modeling and performance evaluation with computer science. Wiley, Hoboken, New Jersey

International Journal of Production Research, 2013
Vol. 51, No. 18, 5576–5596, http://dx.doi.org/10.1080/00207543.2013.789937

Taylor & Francis
Taylor & Francis Group

An integrated operations-marketing perspective for making decisions about order penetration point in multi-product supply chain: a queuing approach

Ebrahim Teimoury and Mahdi Fathi*

Department of Industrial Engineering, Iran University of Science and Technology, Tehran, Iran

(*Received 4 April 2012; final version received 21 March 2013*)

This study is dedicated to strategic decision-making regarding order penetration point (OPP), which is the boundary between make-to-order (MTO) and make-to-stock (MTS) policies. This paper considers a supply chain in which a manufacturer produces semi-finished items on an MTS basis for a retailer that will customise the items based on MTO policy. This two-echelon supply chain offers different products to a market comprised of homogenous customers who have different preferences and willingness to pay. The retailer wishes to determine the optimal OPP, the optimal semi-finished goods buffer size, and the price of the products with assumption of price sensitive demand function. Moreover, we consider both shared and unshared capacity models in this paper. A matrix geometric method is utilised to evaluate various performance measures for this system, and then, optimal solutions are obtained by enumeration techniques. The suggested queuing approach is based on a new perspective between the operations and marketing functions which captures the interactions between several factors including inventory level, product pricing, OPP, and delivery lead time. Finally, parameter sensitivity analyses are carried out and the effect of demand on the profit function, the effect of prices ratio on completion rates ratio and buffer sizes ratio and the variations of profit function for different prices, completion percents, and buffer sizes are examined.

Keywords: queuing system; supply chain; order penetration point (OPP); integrated operations-marketing perspective; MTS-MTO queue; matrix geometric method (MGM)

1. Introduction

One production system which has recently attracted researchers' and practitioners' consideration is hybrid make-to-order/make-to-stock (MTS-MTO) (Rafiei and Rabbani 2012). The MTS production system can meet customer orders fast, but confronts inventory risks associated with short product life cycles and unpredictable demands. In contrast, MTO producers can provide a variety of products and custom orders with lower inventory risks at the expense of longer customer lead times. Moreover, in MTS production, products are stocked in advance, while in MTO, a product starts to be produced only after an order is received. The MTS-MTO supply chain is appropriate wherever common modules are shared by various finished products through divergent finalisation. The MTS-MTO supply chain inherits two key characteristics: First, it can reduce cost of producing standard modules by taking advantage of economies of scale during the MTS stage. Second, it can concurrently satisfy the need for high product variety by taking advantage of MTO's flexibility (Wang et al. 2011). To differentiate the three above-mentioned systems, the concept of Order Penetration Point (OPP) is utilised in Figure 1. This point specifies the stage where the customer's desired specifications influences the production value chain (Hoekstra, Romme, and Argelo 1992). As shown in Figure 1, customer's specifications are taken into consideration in different places along the production systems in MTS, MTO and MTS-MTO.

Delayed product differentiation (DPD) is a common concept in supply chain management in which the manufacturing process starts by making a generic or family product that is later differentiated into a specific end-product. This method is widely used, especially in industries with high demand uncertainty, and can be effectively used to address the final demand even if anticipations cannot be improved. DPD leads to improved customer satisfaction and manufacturing performance through balancing various costs pertaining to different products with different specifications. Besides, as stated by Jewkes and Alfa (2009), DPD can increase a manufacturer's flexibility to deal with uncertainties in market demand. It can be achieved by, for instance, component sharing or reversing the order of operations, as discussed by Lee and Tang (1997). The benefits of DPD as claimed by Jewkes and Alfa (2009) consist of an ability to provide

*Corresponding author. Email: mfathi@iust.ac.ir

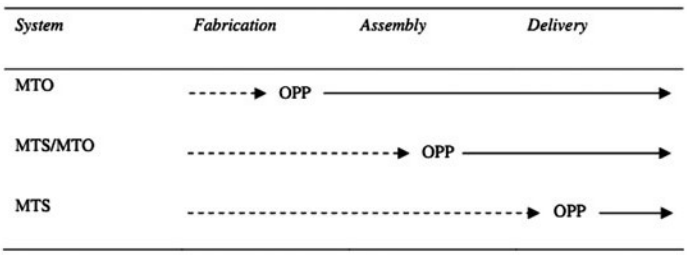


Figure 1. Different production systems; dotted and solid lines represent forecast-driven and customer-order-driven activities, respectively.

custom products with lower customer order fulfilment delay compared to pure MTO systems' delay, and, due to inventory pooling, the ability to hold less overall inventory than MTO systems. The drawbacks of DPD include the potential for increased costs (due to additional material or processing costs), and the risk of greater yield losses or expenditures of process redesign. Another risk of DPD is pertained to the product market – it may become more difficult to respond to the full range of final product specifications demanded by customers.

The positioning of OPP is a challenging area that has received an increasing attention in the manufacturing strategy literature (Hallgren and Olhager 2006). As mentioned in Figure 1, OPP is taken into consideration in different locations along the production systems in MTS, MTO and MTS-MTO. Accordingly, we consider three environments MTS, MTO and MTS-MTO for positioning OPP in supply chain networks as the analysis of the problem is different for each environment. By bringing Table 1, we prefer to display a general overview of our developed OPP models for readers in this section. As shown in Table 1, our developed OPP models in Teimoury et al. (2010), Teimoury et al. (2011) and (Teimoury et al. 2012; Teimoury and Fathi 2012; current research) are in MTS, MTO, and MTS-MTO environment, respectively.

The motivation for this study is that companies are showing an increasing interest in incorporating the OPP as an important input into the strategic design of supply chains. Moreover, making decisions on the price of products in a supply chain with a price sensitive demand function is considered as strategic decision-making with respect to location of the OPP. In practical supply chain management, financial aspects such as the price of a finished product, which has a direct relationship with customer satisfaction, play a vital role. This decision making is affected by different factors such as supply chain configuration and structure, and delivery lead-time. Therefore, we believe that the integrated operations-marketing perspective is needed in positioning OPP in supply chain networks. The rest of the paper is organised as follows. The corresponding literature is reviewed in the next section. The problem description and list of notation are explained in Section 3. The model formulation is studied under shared and unshared inventory capacity in Section 4. Moreover, the queuing aspect and performance evaluation indices are studied. Section 5 is dedicated to a two products supply chain numerical example. And finally, the study is concluded in Section 6.

2. Literature review

There are a number of papers addressing the issue of making decisions on OPP, which has appeared in the literature with various names, such as decoupling point (DP), DPD and product customization postponement. The term DP, in the logistics framework was first introduced by Sharman (1984) where he argued the DP's dependency on a balance between product cost, competitive pressure and complexity.

Positioning OPP includes MTS or MTO decision or hybrid MTS-MTO decision making. According to Shao and Dong (2012) the selection between MTS and MTO is an important decision in many industries, such as contract manufacturers Kumar et al. (2007), plastic toy manufacturing firms Rajagopalan (2002), food companies (Van Donk 2001; Soman, Van Donk, and Gaalman 2004; Akkerman, Van der Meer, and Van Donk 2010), steel mills Kerkkänen 2007, semiconductor plants Chang et al. (2003), timber industry Yáñez et al. (2009) and personal computer manufacturing firms Vidyarthi, Elhedhli, and Jewkes (2009). There is also a large amount of literature explicitly dealing with the hybrid MTO–MTS problem (Sox, Thomas, and McClain 1997; Carr and Duenyas 2000; Soman, Van Donk, and Gaalman 2004; Hallgren and Olhager 2006; Perona, Saccani, and Zanoni 2009; Jewkes and Alfa 2009; Teimoury et al. 2012; Teimoury and Fathi 2012). A comprehensive literature review on MTS-MTO production systems and revenue management of demand fulfilment can be found in Perona, Saccani, and Zanoni (2009) and Quante, Meyr, and Fleischmann (2009).

Table 1. Details of our developed OPP models.

Authors	Problem	Supply chain environment	Demand	Customer type	OPP	Modelling method	Research questions
Teimoury et al. (2010)	A queuing approach to production-inventory planning for supply chain with uncertain demands: Case study of PAKSHOO Chemicals Company	MTS	Poisson process	Multi classes	Located at the end of supply chain network	Stochastic modelling and optimisation, queuing and Markov chain modelling	How can the operational decisions of production-inventory planning be optimised simultaneously in the MTS supply chain environment?Can queuing approach be applied to model uncertainty in demand and delivery time in MTS environment?
Teimoury et al. (2011)	Price, delivery time, and capacity decisions in an M/M/1 make-to-order/service system with segmented market	MTO	Price and delivery lead-time sensitive demand function	Two classes	Located in front of supply chain network		Can queuing approach be applied to model uncertainty in demand and delivery time in MTO environment? According to delivery lead-time and price sensitive demand function of customers in MTO environment, how queuing approach can be applied to optimise price, delivery lead-time and capacity?
Teimoury et al. (2012)	A queuing approach for making decisions about order penetration point in multi-echelon supply chains	MTS-MTO	Poisson process	Multi classes	Decision variable		Can queuing approach be applied to determine the OPP of a multi-product supply chain in MTS-MTO environment? How stochastic modelling and optimisation is proportional to the structure of positioning OPP under uncertain demand and the delivery time? How can the logistics process and transportation mode of finished products to the customers in determining OPP be optimised?
Teimoury and Fathi (2012)	A queuing approach for making decisions about order penetration point in supply chain with impatient customer	MTS-MTO	Poisson process	Single class, impatient customer	Decision variable		Can queuing approach be applied to determine the OPP of a supply chain with impatient customers in MTS-MTO environment?How stochastic modelling and optimization is proportional to the structure of positioning OPP under uncertain demand and the delivery time with impatient customers?Is considering impatient customer important in determining OPP in MTS-MTO environment?

(*Continued*)

Table 1. (*Continued*).

Authors	Problem	Supply chain environment	Demand	Customer type	OPP	Modelling method	Research questions
Current research	An integrated operations-marketing perspective for making decisions about order penetration point in multi-product supply chain: A queuing approach	MTS-MTO	Price sensitive demand function	Multi classes	Decision variable		Can queuing approach be applied to determine the OPP of a multi-product supply chain with an integrated operations-marketing perspective in MTS-MTO environment?How stochastic modelling and optimisation is proportional to the structure of positioning OPP under uncertain demand and the delivery time with an integrated operations-marketing perspective?How can OPP be determined based on an integrated operations-marketing perspective with the assumption of price sensitive demand function?

Adan and Van der Wal (1998) studied the effect of MTS and MTO production policies on order satisfaction lead-times. Arreola-Risa and DeCroix (1998) analysed the effect of manufacturing-time diversity on MTO/MTS decisions and presented optimality conditions for MTO/MTS partitioning in a multi-product, single-machine case with an FCFS scheduling rule. Their results showed the extent to which reducing manufacturing-time randomness leads to MTO production. Recently, Günalay (2011) studied the efficient management of MTS or MTO production-inventory system in a multi-item manufacturing facility. Rajagopalan (2002) proposed a model and a solution approach for deciding whether a set of items should be MTS or MTO and the production policy for the MTS items. The objective of his model was to minimise inventory costs of MTS items while ensuring that orders for MTO items were satisfied within a lead time, T, with a specified probability. Su et al. (2010) analysed the cost and benefit of implementing DPD in an MTO environment (in the Hewlett-Packard printer case, printers were made in an MTS environment) by means of queuing models.

The trade-off between aggregation of inventory (or inventory pooling) and the costs of redesigning the production process is studied by Aviv and Federgruen (2001) where congestion impacts are not taken into account. In contrast, Gupta and Benjaafar (2004) included the impact of capacity restrictions and congestion, i.e. they proposed a common framework to examine MTO, MTS and DPD systems in which production capability is considered. Furthermore, they analysed the optimal postponement point in a multi-stage queuing system. The DPD issue in manufacturing systems is studied by Jewkes and Alfa (2009) in which they decided on where to locate the point of differentiation in a manufacturing system, and also what size of semi-finished products inventory storage should be considered. In addition, they presented a model to realise how the degree of DPD affects the trade-off between customer order completion postponement and inventory risks, when both stages of production have non-negligible time and the production capacity is limited. Their model did not, however, consider the demand to be a function of price. In this paper, we extend their model for multi-product supply chain under shared and unshared inventory capacity and consider the demand to be a function of price products. Such an extension is useful in viewing the problem in an integrated operations-marketing perspective which is more practical for managers.

Recently, Ahmadi and Teimouri (2008) studied the problem of where to locate the OPP in an Auto Export supply chain by means of dynamic programming. Teimoury et al. (2010) proposed an integrated two stage inventory-queue model and production planning model based on queuing approach in real case study of PAKSHOO chemicals company uncertain demands. Teimoury and Fathi (2012) developed a queuing model for locating OPP in a two-echelon supply chain with impatient customers. While Teimoury et al. (2012) proposed a queuing model for making decisions about OPP in multi-echelon supply chains, they did not discuss in an integrated operations-marketing framework. Furthermore, a notable literature review in positioning DPs and multiple DPs in a supply network can be seen in Sun et al. (2008); these positioning models did not, however, make any decisions about the optimal semi-finished buffer size and optimal fraction of processing time fulfilled by the upstream of DP. Wong, Wikner, and Naim (2009) studied postponement based on the positioning of the differentiation points and the stocking policy. Wee and Dada (2010) studied a make-to-stock manufacturing system with component commonality based on queuing approach. Jeong (2011) developed a dynamic model to simultaneously determine the optimal position of the decoupling point and production–inventory plan in a supply chain.

This paper investigates an integrated operations-marketing perspective based on queuing approach for making decisions about OPP in supply chain. A comprehensive review of operations-marketing interface models is studied by Tang (2010) and many applications and methods of operations-marketing perspective are surveyed in O'Leary-Kelly and Flores (2002), Ho and Zheng (2004), Ray (2005), Feng, D'Amours, and Beauregard (2008), Ioannidis and Kouikoglou (2008), Rao (2009), Vandaele and Perdu (2010), Feng, D'Amours, and Beauregard (2010), Erickson (2011), Oliva and Watson (2011), Wong and Eyers (2011) and Chayet, Hopp, and Xu (2004). Many applications and methods for determining the OPP are also presented in Olhager (2003, 2010), Yang and Burns (2003), Mikkola and Skjøtt-Larsen (2004), Yang, Burns, and Backhouse (2004), Rudberg and Wikner (2004), Wikner and Rudberg (2005), Skipworth and Harrison (2004, 2006), Harrison and Skipworth (2008), Wong, Wikner, and Naim (2009), Banerjee, Sarkar, and Mukhopadhyay 2012, and Choi, Narasimhan, and Kim (2012). Moreover, following authors have developed their models based on queuing approach (Arreola-Risa and DeCroix 1998; Gupta and Benjaafar 2004; Wong, Wikner, and Naim 2009; Jewkes and Alfa 2009; Wee and Dada 2010; Su et al. 2010; Wong, Wikner, and Naim 2010; Teimoury et al. 2010; Wong and Eyers 2011; Teimoury et al. 2011; Teimoury et al. 2012; Teimoury and Fathi 2012). According to Table 2, we are the authors of five papers out of 13 available papers in the literature and current work is based on highlighted papers in Table 2 which are mostly related to the literature of our study.

For the first time, current research based on queuing approach considers pricing decisions, determining decoupling point and warehouse capacity planning, simultaneously. Having added the pricing decision and assumption of price sensitive demand function to the developed model by Teimoury et al. (2012), the proposed model improves the OPP model and makes it more realistic and applicable to real cases. The mathematical models in OPP literature commonly seek a

Table 2. Literature review of developed OPP models based on queuing approach.

Authors	Problem	Published journal
Arreola-Risa and DeCroix (1998)	Make-to-order versus make-to-stock in a production–inventory system with general production times	*IIE Transactions*
Gupta and Benjaafar (2004)	Make-to-order, make-to-stock, or delay product differentiation? A common framework for modelling and analysis.	*IIE Transactions*
Wong, Wikner, and Naim (2009)	Analysis of form postponement based on optimal positioning of the differentiation point and stocking decisions	*International Journal of Production Research*
Jewkes and Alfa (2009)	A queuing model of delayed product differentiation	*European Journal of Operational Research*
Wee and Dada (2010)	A make-to-stock manufacturing system with component commonality: A queuing approach	*IIE Transactions*
Su et al. (2010)	The impact of delayed differentiation in make-to-order environments	*International Journal of Production Research*
Wong, Wikner, and Naim (2010)	Evaluation of postponement in manufacturing systems with non-negligible changeover times	*Production Planning & Control*
Wong and Eyers (2011)	An analytical framework for evaluating the value of enhanced customisation: an integrated operations-marketing perspective.	*International Journal of Production Research*
Teimoury et al. (2010)	A queuing approach to production-inventory planning for supply chain with uncertain demands: Case study of PAKSHOO Chemicals Company	*Journal of Manufacturing Systems*
Teimoury et al. (2011)	Price, delivery time, and capacity decisions in an M/M/1 make-to-order/ service system with segmented market	*International Journal of Advanced Manufacturing Technology*
Teimoury et al. (2012)	A queuing approach for making decisions about order penetration point in multi-echelon supply chains	*International Journal of Advanced Manufacturing Technology*
Teimoury and Fathi (2012)	A queuing approach for making decisions about order penetration point in supply chain with impatient customer	*International Journal of Advanced Manufacturing Technology*
Current research	An integrated operations-marketing perspective for making decisions about order penetration point in multi-product supply chain: A queuing approach	*International Journal of Production Research*

balance between inventory costs and customer service levels, but to the authors' knowledge, pricing problem in OPP positioning has not been noticed in literature nonetheless. The proposed queuing approach is based on a new perspective between the operations and marketing functions which captures the interactions between several factors including inventory level, price, OPP, and delivery lead time. Moreover, the proposed model attempts to maximise the revenue of the supply chain. Therefore, the model should optimise the price of each product type which results in an integrated operations-marketing interface perspective which has become more practical and more comprehendible to supply chain managers.

The goal of this paper is to find equilibrium customer service levels with inventory costs, as akin to developed models in the literature as in Teimoury et al. (2010, 2011, 2012), Teimoury and Fathi (2012) and Jewkes and Alfa (2009). Ours, however, differs from the studied articles in several ways. First, pricing decision making is added to the OPP model as a decision variable. Second, the literature chiefly focuses on single product modelling. On the contrary, the developed model covers multi-product supply chains. Third, in contrast to the previous studies in literature, we assume, for the first time, a price sensitive demand function in our OPP positioning model and demand function is considered to be a function of prices of different products which are replaceable. Finally, in this study, there are: practical base; integrating operations-marketing perspective by adding decision on product pricing with assumption of price sensitive demand function and theoretical base; applying queuing approach for modelling the problem because of uncertain nature of demand arrival and lead-time. Therefore, this model optimises both marketing and operations simultaneously to obtain OPP in supply chain networks and our point of view to the problem helps our previous OPP positioning models to get closer to practical models in real cases.

The supply chain which is considered as a basic model in this paper is composed of two production stages. In the first production stage, the manufacturer produces semi-finished products on an MTS policy for a retailer in the second production stage that will customise the products based on an MTO policy. The semi-finished products will be completed as a result of specific customer orders. The developed model obtains the optimal prices of the products for the completed products to each demand point. In order to balance the costs of customer order fulfilment delay and inventory

costs of each product type, retailer tries to find the optimal fraction of processing performed by the manufacturer and its optimal semi-finished products buffer storage.

3. Problem description and list of notation

The following notations are used for the integrated operations-marketing mathematical formulation of the proposed model.

Sets and indices:

m_i Semi-finished products buffer storage capacity for product of type i index $m_i = 1, 2, ..., S_i$

i Product's type index $i = 1, 2, \ldots, L$

Decision variables:

θ_i Percent of completion for product of type i in the first production stage

S_i Storage capacity of type i semi-finished products

P_i Price quoted to product of type i

Parameters:

$V(\theta_i)$ The value per unit of semi-finished products (dollar/unit)

τ_i Constant fraction of the MTO processing rate for product of type i

μ_i Mean production rate for product of type i

C_{H_i} The holding cost for semi-finished product of type i (dollar/unit)

C_{W_i} The cost of customer order fulfilment delay for product of type i (dollar/unit)

C_{C_i} The cost of establishing type i semi-finished products storage capacity (dollar/unit)

C_{u_i} The cost of disposing an unsuitable item of type i (dollar/unit)

λ_i Mean arrival rate for product of type i

Expected performance measures:

$E(N_i)$ The expected number of type i semi-finished products in the system

$E(W_i)$ The expected customer order completion delay for product of type i – the time from when a customer order enters the system until its product is completed

$E(U_i)$ The expected number of type i unsuitable products produced per unit time

A production supply chain is considered in which a manufacturer produces semi-finished items on a MTS basis for a retailer as shown in Figure 1. Customer orders for completed products arrive at the retailer and are filled on a MTO basis by customising the semi-finished goods to customer specifications. It is assumed that the studied supply chain offers L products to a market comprising homogenous customers that differ in their preferences for willingness to pay. It is considered that a retailer is dealing with multiple types of customers who have different Poisson demand rate and are sensitive to price of the requested product and other products in the line. The demands are differentiated based on the products. According to Tsay and Agrawal (2000), Boyaci and Ray (2007) and Teimoury et al. (2011), the demand rates are modelled using the linear functions $\lambda_i = \alpha_i - \beta_i P_i + \sum_{j \neq i}^{L} \gamma_{ij} P_j$. The demands for a two-product supply chain are as follows:

$$\lambda_1 = \alpha_1 - \beta_1 P_1 + \gamma_1 P_2 \tag{1}$$

$$\lambda_2 = \alpha_2 - \beta_2 P_2 + \gamma_2 P_1 \tag{2}$$

The proposed model seeks to maximise the revenue of the supply chain. Therefore, the model should optimise the price of each product type and this makes an integrated operations-marketing interface perspective.

In this system, customers arrive at random times and each customer requests one unit of a product. The times between successive customer arrivals are independent random variables with rate λ in accordance to a Poisson process. It is assumed that each customer orders one unit of type-i product with a probability of q_i where $\sum_{i=1}^{L} q_i = 1$ and $\lambda_i = \lambda q_i$, $i = 1, 2, ..., L$. The production times of work stations for all product types are assumed to be exponentially distributed with rates μ_i, $i = 1, 2, ..., L$ where $\sum_{i=1}^{L} \mu_i = \mu$. Moreover, it is supposed that the manufacturer has an infinite source of raw materials and never faces shortage. The second production stage has to determine the optimal storage capacity of type i semi-finished products (S_i, $i = 1, 2, ..., L$). Figure 2a and Figure 2b illustrate a diagram depicting the model under both shared (Section 4.1) and unshared (Section 4.2) capacity models, respectively.

As shown in Figure 2, the manufacturer provides undifferentiated semi-finished products to the final production stage. For each product type, manufacturer produces a semi-finished product ($\%100\theta_i$ completed ($0 < \theta_i < 1$)) to be delivered to the final production stage. The final production stage then completes the remaining $1 - \theta_i$ fraction according to a particular customer order. It should be noted that the manufacturer is not necessarily in a different organisation from the retailer; the 'manufacturer' and 'final production stage' may be two successive stages in a same organisation. We modelled θ_i as a continuous variable in order to gain profound insights into the overall relationship between θ_i and the performance of the system. The assumption also facilitates our computational analysis. Therefore, the results is presented as if the final production stage can implement any values of θ_i. If this is not the case, our model enables us to quickly identify the best choice of θ_i among a finite number of feasible alternatives. According to market characteristics

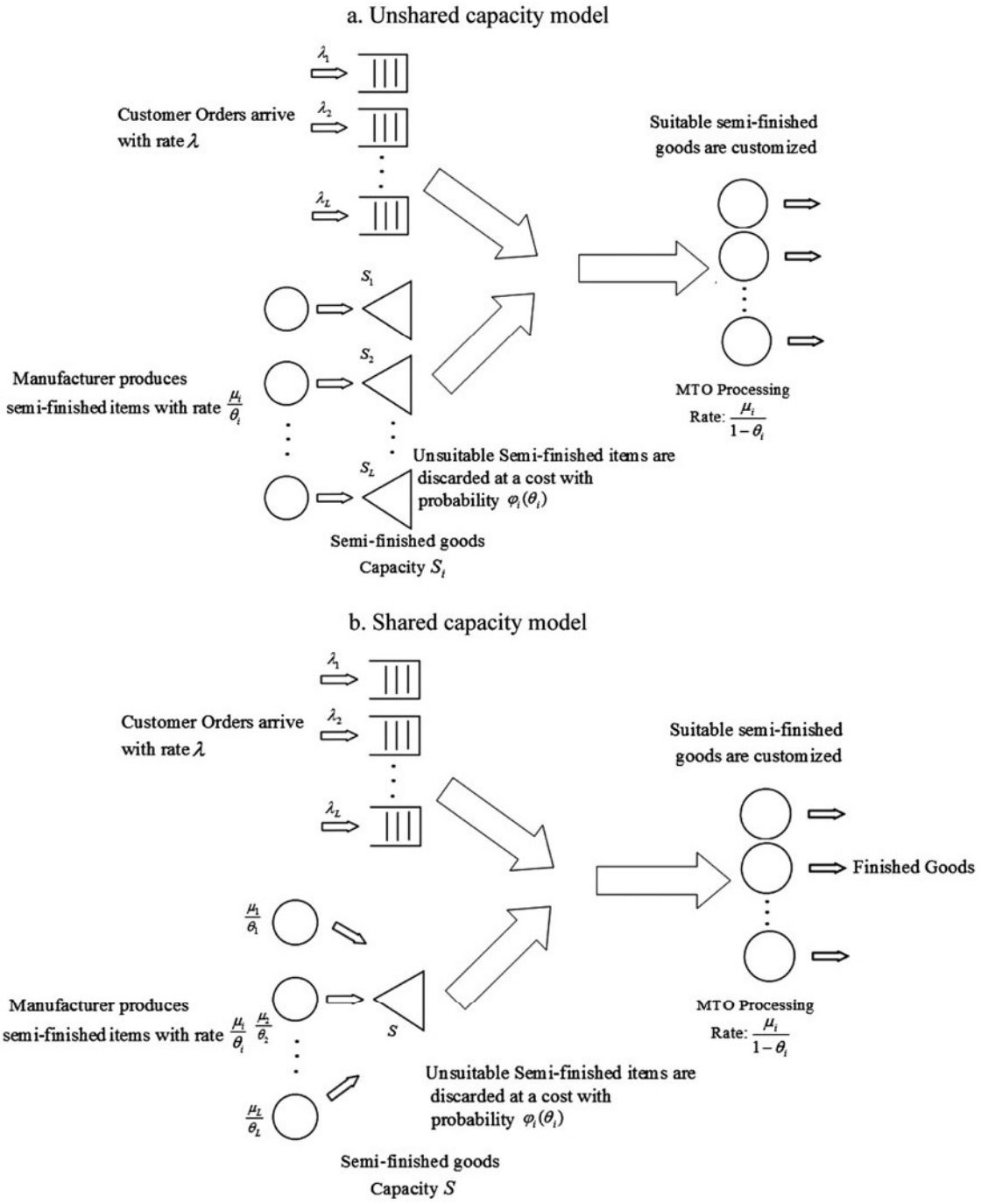


Figure 2. The multi-product hybrid MTS-MTO production supply chain.

studied by Jewkes and Alfa (2009), there is a probability of $\phi_i(\theta_i)$ that a semi-finished product is not suitable for customisation and so $\phi_i(\theta_i)$ is monotonically increasing with θ_i which is a reasonable assumption. The value φ_i can be thought of as a characteristic of the product marketplace. High values of φ_i represent a marketplace for which high degrees of customisability is important to consumers. Low values of φ_i represent a market place in which customisation is less important to customers. In terms of a mathematical representation for φ_i, we may assume, for example, that $\varphi_i = b_i\theta_i^n$, $n \geq 1$; $0 < b_i < 1$. More general forms can be modelled. For the time being, however, we will assume $n = 1$, i.e. $\varphi_i = b_i\theta_i$. A practical value of b_i will depend on characteristics of the customer population. High values of b_i (close to 1.0) means that the market demands a high degree of freedom to specify the final product and is intolerant to deviation. Lower values of b_i might be appropriate if customers will accept a range of product characteristics – i.e. there is a smaller probability that the item will be unsuitable even if it has characteristics stemming from DPD (Jewkes and Alfa 2009).

4. Problem formulation

4.1 *Unshared capacity model*

The entire explained system for unshared capacity model, which is shown in Figure 2(a) in Section 3, can be described by a Markov process with state (n_i, m_i), where n_i is the number of customers in the system waiting for each finished product of type i and m_i is the number of type i semi-finished products in its semi-finished product storage. Therefore, the state space is denoted by $\Omega = \{n_i \geq 0,\ 0 \leq m_i \leq S_i\}$, which is depicted in Figure 3 with transition rates.

In Figure 3, for each product type $a = \frac{\mu(1-\phi)}{\theta}$ and $b = \frac{\mu}{1-\theta}$. The associated balance equations for the steady probabilities follow Equations (3) to (8).

$$\left(\frac{\mu_i(1-\varphi_i)}{\theta_i} + \lambda_i\right) P_i(n_i, m_i) = \frac{\mu_i}{1-\theta_i} P_i(n_i + 1, m_i + 1), \qquad n_i = 0, m_i = 0 \tag{3}$$

$$\left(\frac{\mu_i(1-\varphi_i)}{\theta_i} + \lambda_i\right) P_i(n_i, m_i) = \frac{\mu_i(1-\varphi_i)}{\theta_i} P_i(n_i, m_i - 1) + \frac{\mu_i}{1-\theta_i} P_i(n_i + 1, m_i + 1), \quad n_i = 0, 1 \leq m_i \leq S_i - 1 \tag{4}$$

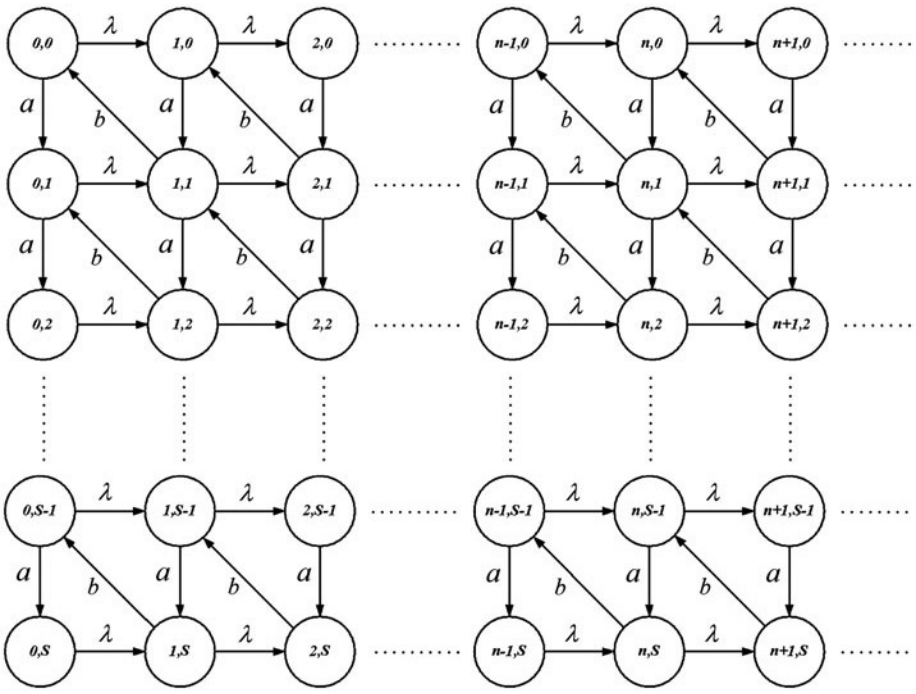


Figure 3. State transition rates diagram.

$$\frac{\mu_i(1-\phi_i)}{\theta_i}P_i(n_i\,,m_i-1)=\lambda_i\,P_i(n_i,m_i),\quad n_i=0\,,m_i=S_i \tag{5}$$

$$\left(\frac{\mu_i(1-\varphi_i)}{\theta_i}+\lambda_i\right)P_i(n_i\,,m_i)=\lambda_i\,P_i(n_i-1\,,m_i)+\frac{\mu_i}{1-\theta_i}P_i(n_i+1\,,m_i+1),\quad 1\le n_i\,,m_i=0 \tag{6}$$

$$\left(\frac{\mu_i(1-\varphi_i)}{\theta_i}+\lambda_i+\frac{\mu_i}{1-\theta_i}\right)P_i(n_i\,,m_i)=\frac{\mu_i(1-\varphi_i)}{\theta_i}P_i(n_i\,,m_i-1)+\frac{\mu_i}{1-\theta_i}P_i(n_i+1\,,m_i+1)+\lambda_i\,P_i(n_i-1\,,m_i),\quad n_i=0,\quad 1\le m_i\le S_i-1 \tag{7}$$

$$\left(\lambda_i+\frac{\mu_i}{1-\theta_i}\right)P_i(n_i\,,m_i)=\frac{\mu_i(1-\varphi_i)}{\theta_i}P_i(n_i\,,m_i-1)+\lambda_i\,P_i(n_i-1\,,m_i),\quad 1\le n_i,\quad m_i=S_i \tag{8}$$

The corresponding generator matrix Q_i written in block form (9) for the product of type i is:

$$Q_i=\begin{bmatrix} D_i & A_i & & & \\ C_i & E_i & A_i & & \\ & C_i & E_i & A_i & \\ & & \ddots & \ddots & \ddots \end{bmatrix} \tag{9}$$

Appendix A shows block matrices where A_i, C_i, E_i and G_i are block matrices with the dimension of $(S_i+1)\times(S_i+1)$. It is notable that A_i giving the rate at which the number of customer orders in the system increases by one, E_i giving the rate at which the number of customer orders in the system either stays at the same level and C_i giving the rate at which the number of customer orders in the system decreases by one. G_i is the matrix rate at which the customer orders in the system move from zero to one.

Let $F_i=A_i+E_i+C_i$ be a generator matrix with its associated stationary distribution $P_i=[P_{i0},P_{i1},...,P_{iS_i}]$ given as a solution to $P_iF_i=0$, $P_i\mathbf{1}=1$.

$$F_i=\begin{bmatrix} F_{i_{0,0}} & F_{i_{0,1}} & & & \\ F_{i_{1,0}} & F_{i_{1,1}} & F_{i_{1,2}} & & \\ & \ddots & \ddots & \ddots & \\ & & F_{i_{S_i-1,S_i-2}} & F_{i_{S_i-1,S_i-1}} & F_{i_{S_i-1,S_i}} \\ & & & F_{i_{S_i,S_i-1}} & F_{i_{S_i,S_i}} \end{bmatrix} \tag{10}$$

Appendix B illustrates block matrices where $F_{i_{m,m+1}}$, $F_{i_{m,m-1}}$, and $F_{i_{m,m}}$ are $(S_i+1)\times(S_i+1)$. As it is discussed in Neuts (1981), the explained Markov chain is stable if $P_iC_i\mathbf{1}>P_iA_i\mathbf{1}$. In order to have a stable system, we require the final production stage to have a service rate that exceeds the arrival rate of customers. In addition, the supply rate of suitable semi-finished products to the final production stage must be more than the customer demands rate.

4.1.1 *Steady state analysis*

The behaviour of this supply chain system is studied in a steady state. Let $\Pi_i=[\Pi_{i\mathbf{0}},\Pi_{i\mathbf{1}},\Pi_{i\mathbf{2}},...]$ be the stationary probabilities associated with the Markov chain for each product type so that $\Pi_iQ_i=0$ and $\Pi_i\mathbf{1}=1$ $(i=1,2)$. Due to the matrix geometric theorem Neuts (1981), equation $\Pi_{i,n+1}=\Pi_{i,n}R_i,\quad n\ge 0$ must be satisfied where R_i is the minimal non-negative solution to the matrix quadratic equation $A_i+R_iE_i+R_i^2C_i=0$.

It is noteworthy that matrix R_i can be computed very easily using some well known methods according to Bolch et al. (1998). A simple way to compute R_i is the iterative approach given as $R_i(n+1)=-(A_i+R_i(n)^2C_i)E_i^{-1}$ until $|R_i(n+1)-R_i(n)|_{nj}<\varepsilon$, with $R_i(0)=0$. The boundary vector Π_{i0} is obtained from $\Pi_{i0}(D_i+R_iC_i)=0$.

4.1.2 *Performance evaluation indices*

Here, the important performance evaluation indices of the system can be obtained as described below. Let $E[O_i]$ be the mean number of customers' orders for product of type i in the system, including the one being served; $E[W_i]$ be the mean customer order completion delay for product of type i; $E[N_i]$ be the mean number of semi-finished products in the system for product of type i, and $E[U_i]$ be the expected number of unsuitable semi-finished products disposed per unit time for product of type i, then

$E[O_i] = \Pi_{i1}(I - R_i)^{-2}\mathbf{1}$

$E[W_i] = \frac{E(O_i)}{\lambda_i}$ (by applying Little's Law),

$E[N_i] = \Pi_{i0}(I - R_i)^{-1}y_i$; where $y_i = [0, 1, 2, ..., S_i]^T$,

$E(U_i) = \dfrac{(1 - \Pr(m_i = S_i))\varphi_i\mu_i}{\theta_i}$; where m_i denotes the number of semi-finished products storage for each product type.

4.1.3 *Mathematical model*

The objective function includes the following costs:

(1) Holding semi-finished products in buffer storage (C_{H_i}).
(2) Establishing semi-finished products storage capacity (C_{C_i}).
(3) Customer order fulfilment delay (C_{W_i}).
(4) Disposing an unsuitable item (C_{u_i}).

The integrated operations-marketing mathematical formulation of the model is as follows:

$$\underset{P_i,S_i,\theta_i}{Max}\ Z(P_i, S_i, \theta_i) = \sum_{i=1}^{L} P_i\lambda_i - \sum_{i=1}^{L} C_{u_i}V(\theta_i)E(U_i) - \sum_{i=1}^{L} C_{h_i}V(\theta_i)E(N_i) - \sum_{i=1}^{L} C_{w_i}E(W_i) - \sum_{i=1}^{L} C_{c_i}S_i \tag{11}$$

St:

$$\frac{(1 - \theta_i)}{\mu_i} \geq \tau_i E(W_i) \quad \forall i \tag{12}$$

$$\lambda_i \geq 0 \quad \forall i \tag{13}$$

$$0 < \theta_i < 1.0 \quad \forall i \tag{14}$$

$$S_i = 1, 2, ... \quad \forall i \tag{15}$$

$$P_i \geq 0 \quad \forall i \tag{16}$$

The objective function (11) maximises the total expected profit in the supply chain. The cost structure consists of the cost of semi-finished products that are not consistent with customer's order, expected semi-finished products' holding cost, the cost of establishing storage capacity for semi-finished products, and expected cost of delay in customer order completion which include time of customisation and logistics. According to Jewkes and Alfa (2009), the second production stage wishes to impose a service level constraint to limit the expected customer order fulfilment delay to a set threshold. Empirical studies show that order processing time is typically about 5–20% of order lead time; hence, the second production stage establishes the service level threshold in relation to the average amount of time spent customising a semi-finished item. Therefore, constraint (12) is employed for each product type $\left(\frac{(1-\theta_i)}{\mu_i} \geq \tau_i E(W_i)\right)$. In other words, the mean time it takes for the manufacturer to customise the order, $\frac{\mu_i}{(1-\theta_i)}$, must be at least a fraction τ_i of the overall customer order fulfilment delay. Values of τ are considered in the range of $0.05 \leq \tau_i \leq 0.20$. Constraints (13) and (16) restrict the value of mean arrival rate and price for product of type i to be non-negative. Constraint (14) assures

that the percent of completion for product of type i in first production stage is between zero and one. The constraint (15) represents the range of the storage capacity of type i semi-finished products.

The outputs of the represented model are the optimal fractions of the process fulfilled by the manufacturer for each product type, optimal storage capacity of each semi-finished product, and the optimal prices for each product type.

4.1.4 *Solution approach*

Based on the fact proposed in constraint (13) the value of mean arrival rate for product of type i must be non-negative. Considering this constraint, there is an upper bound and a lower bound for each product's price which can be simply determined.

In order to be able to solve Markov-related section of the problem, it is necessary to have the specific amount of λ_i. In this case, it is needed to calculate the amount of λ_i for different values of prices. For the discrete values of prices distributed from the introduced upper bound and lower bound, different values of λ_i are calculated. Then for each λ_i, stochastic values of the objective function Z are calculated with the help of matrix geometric method. The final model will be solved by means of stochastic search in order to specify the optimal values of S_i, θ_i pertaining to the optimal profit function. A numerical example will be proposed in Section 5 to illustrate the function of this solution approach.

4.2 *Shared capacity model*

This section studies a more realistic case that can be considered as a supplement to the proposed model (see Figure 2(b)). According to warehouses physical structure, we cannot establish every calculated optimal storage capacity for each product type. This is a cogent assumption in operational problems. Moreover, specific capacity of S for semi-finished product warehouse is considered. Due to separate calculations of optimal storage capacity for each product type, we cannot apply the storage space constraint in our optimization model. Therefore, if the cumulative semi-finished product storage for all types of products satisfies the warehouse capacity constraint, the obtained solutions can be taken into consideration as optimal storage capacities for products. On the contrary, if the warehouse capacity constraint has not been satisfied, according to Teimoury et al. (2012) we can use the developed heuristic solution procedure as follows.

Algorithm

Step 1: Set $S_0 = (S_1, S_2, ..., S_i, ..., S_L)$ and $Z_0 = Z(P^*(S_0), S_0, \theta^*(S_0))$.

Step 2: Calculate $\sum S_0$ (cumulative storage value for all product types). If $\sum S_0 \leq S$ (S is the predefined capacity constraint for central warehouse), solutions obtained in step 1 are acceptable: stop and set $Z_0 \rightarrow Z^*$; $P^*(S_0) \rightarrow P^*$; $S_0 \rightarrow S^*$; $\theta^*(S_0) \rightarrow \theta^*$. Otherwise: Step 3.

Step 3: Set $Z_0 \rightarrow Z^{MAX}$; $P^*(S_0) \rightarrow P^{MAX}$; $S_0 \rightarrow S^{MAX}$; $\theta^*(S_0) \rightarrow \theta^{MAX}$.

Step 4: Set $Z^{MAX} = Z\left(P^*\left(S^{MAX}\right), S^{MAX}, \theta^*\left(S^{MAX}\right)\right)$

Step 5: Set $S^{MAX} - 1 \rightarrow S^{MAX}$ (if $S^{MAX} - 1$ is stable), $Z_i = Z\left(P^*\left(S^{MAX} - 1\right), S^{MAX} - 1, \theta^*\left(S^{MAX} - 1\right)\right)$ for each product type and solve $\underset{i}{Max}\left(Z_i - Z^{MAX}\right)$.

Step 6: If $\sum S_{i^*} \leq S$, solutions obtained in step 3 are acceptable: stop and set $Z_{i^*} \rightarrow Z^*$; $P^*(S_{i^*}) \rightarrow P^*$; $S_{i^*} \rightarrow S^*$; $\theta^*(S_{i^*}) \rightarrow \theta^*$. Otherwise: set $Z_{i^*} \rightarrow Z^{MAX}$; $P^*(S_{i^*}) \rightarrow P^{MAX}$; $S_{i^*} \rightarrow S^{MAX}$; $\theta^*(S_{i^*}) \rightarrow \theta^{MAX}$ and go to step 3.

The proposed algorithm is represented schematically in Figure 4.

Although the developed algorithm is so time-consuming due to the enumeration technique used in its steps, it computes a nearly optimal solution with minimum benefit loss.

5. Numerical example

We developed the theoretical model in generic terms. In order to apply our model to a real case study, a motor production supply chain network is studied containing two product types with one manufacturer, one retailer, a capacitated warehouse with the capacity of $S = 5$. It is assumed that the demand functions of each product would be as follow.

$$\lambda_1 = 0.2 - 0.05P_1 + 0.01P_2 \geq 0$$

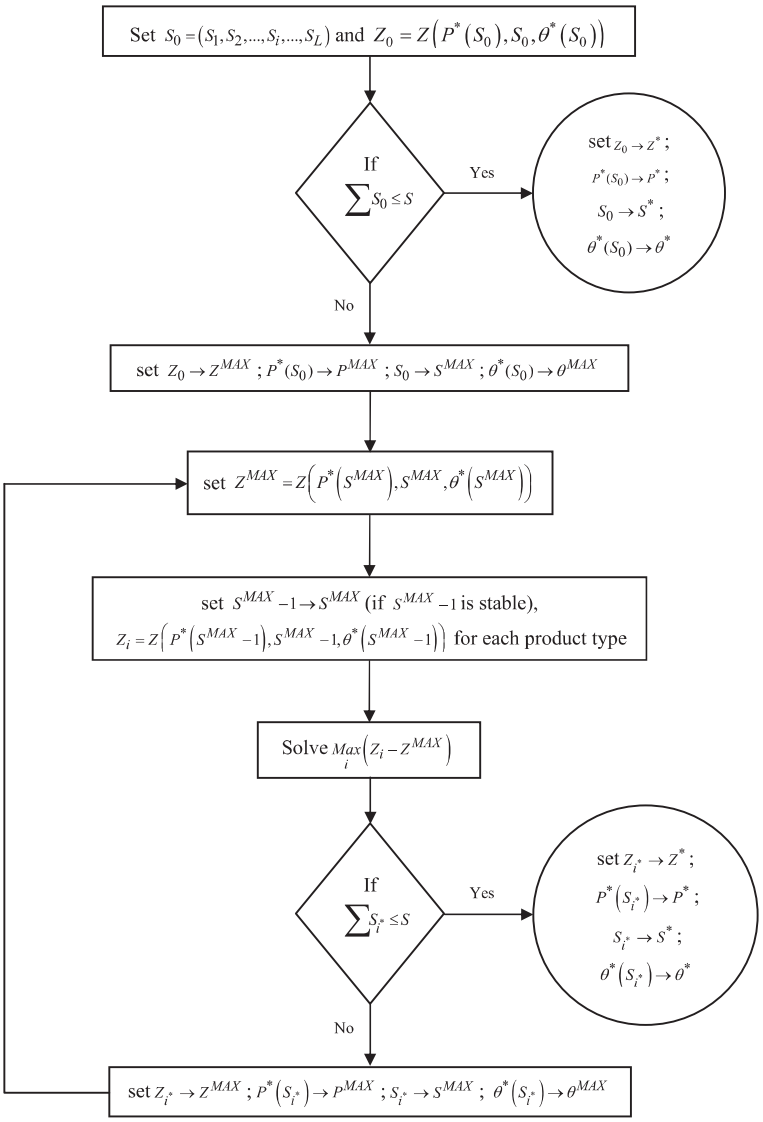


Figure 4. Heuristic solution procedure.

$$\lambda_2 = 0.2 - 0.01P_2 + 0.005P_1 \geq 0$$

Based on the assumed parameters, the feasible solutions of the prices can be calculated easily. Furthermore, each semi-finished product value $V(\theta_i)$ equals to θ_i as assumed by Jewkes and Alfa (2009). Parameters' settings for numerical example, based on the data derived from a real supply chain with two types of motor engines, are:

Table 3. Parameters setting.

	Product *I*	Product *II*
μ_i	0.8	0.7
C_{Ui}	0.0001	0.001
C_{Hi}	0.00001	0.001
C_{Wi}	0.01	0.1
C_{Ci}	0.000005	0.0005
τ_i	0.05	0.05

The integrated operations-marketing mathematical formulation of two product supply chain is as follows:

$$\underset{P_1,P_2,S_1,S_2,\theta_1,\theta_2}{Max} \; Z(P_1,P_2,S_1,S_2,\theta_1,\theta_2) = P_1(\alpha_1-\beta_1P_1+\gamma_1P_2)+P_2(\alpha_2-\beta_2P_2+\gamma_2P_1) - C_{u1}V(\theta_1)E(U_1)-C_{u2}V(\theta_2)E(U_2)-C_{h1}V(\theta_1)E(N_1) - C_{h2}V(\theta_2)E(N_2)-C_{w1}E(W_1)-C_{w2}E(W_2)-C_{c1}S_1-C_{c2}S_2$$

St:

$$\frac{(1-\theta_1)}{\mu_1} \geq \tau_1 E(W_1)$$

$$\frac{(1-\theta_2)}{\mu_2} \geq \tau_2 E(W_2)$$

$$\alpha_1-\beta_1P_1+\gamma_1P_2 \geq 0$$

$$\alpha_2-\beta_2P_2+\gamma_2P_1 \geq 0$$

$$0<\theta_1<1$$

$$0<\theta_2<1$$

$$P_1 \geq 0$$

$$P_2 \geq 0$$

$$S_1 = 1,2,...$$

$$S_2 = 1,2,...$$

The computational results are based on the MATLAB 7.8 implementation where the total cost is computed for $0.01 \leq \theta_i \leq 0.99$ in increments of 0.01 where S_i varies from 1 to 50.

As shown in Table 4, the most beneficial policy is a combination of completing product of type *I* up to 14% based on the predictions and producing the remaining 86% based on the certain demand arrival in the second level, and manufacturing product of type *II* up to 11% based on the predictions and completing 89% based on the certain demand arrival in the second level. In this scenario, a warehouse with capacity of three for product *I* and a warehouse with capacity of two for product *II* are to be established. Moreover, the optimal price for product *I* would be four and for product *II* would be 13. In practice though, these percentages will be adapted to the most conceivable form of product. Furthermore, owing to the low optimal percent of production, it is inevitable that this chain is inclined to produce MTO products. This is conscionably justifiable inasmuch as the cost of disposing an unsuitable item is exorbitant which leads to abrupt reduction in profit function.

The warehouse capacity of ($S = 5$) has to be satisfied in the studied example. Therefore, the satisfaction condition $\sum_{i=1}^{2} S_i^* \leq S$ must be checked and if the storage capacity constraint does not hold, the developed heuristic solution should be implemented:

Table 4. Results of numerical example.

P_1	P_2	θ_1	θ_2	S_1	S_2	$Z(P_1, P_2, \theta_1, \theta_2, S_1, S_2)$
1	1	0.10	0.05	1	1	**0.152512171**
1	1	0.10	0.05	1	2	**0.154758246**
1	1	0.10	0.05	1	3	**0.154271065**
1	1	0.10	0.05	1	4	**0.153759719**
1	1	0.10	0.05	1	5	**0.153248809**
…	…	…	…	…	…	…
4	13	0.14	0.11	3	1	**1.512115610**
4	13	0.14	0.11	3	2	**1.514182834**
4	13	0.14	0.11	3	3	**1.513703482**
4	13	0.14	0.11	3	4	**1.513194496**
4	13	0.14	0.11	3	5	**1.512686447**
…	…	…	…	…	…	…
8	24	0.20	0.15	5	1	**0.160320001**
8	24	0.20	0.15	5	2	**0.159817378**
8	24	0.20	0.15	5	3	**0.159316865**
8	24	0.20	0.15	5	4	**0.158816761**
8	24	0.20	0.15	5	5	**0.158316740**

Step 2: $\sum_{i=1}^{2} S_i^* = 3 + 2 = S$

Therefore, the optimal solution can be taken into account as an optimal solution for the shared capacity case either.

Further analysis of profit function is conducted based on the different measures of price, completion percent, and buffer size. Interrelations between these factors are also investigated. Furthermore, the sensitivity analysis of the parameters is performed by comparing the results of two products as follows.

- Variations of profit function for different prices, completion percentages, and buffer sizes

As shown in different parts of Figure 5, increase in price, completion rate, and buffer size, even though it causes an increase in profit, pursues a decrease due to the increase in waiting costs. This leads to the conclusion that there is a maximum value for the percent of completion for the product of type i in first production stage (θ_i), optimal storage capacity of type i semi-finished products (S_i), and price quoted to product of type i(P_i).

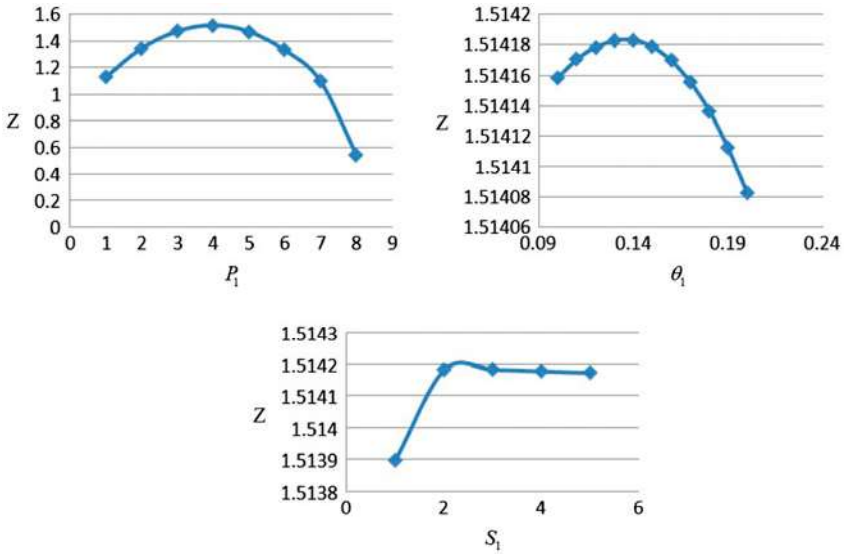


Figure 5. Variations of profit function for different prices, completion percents, and buffer sizes.

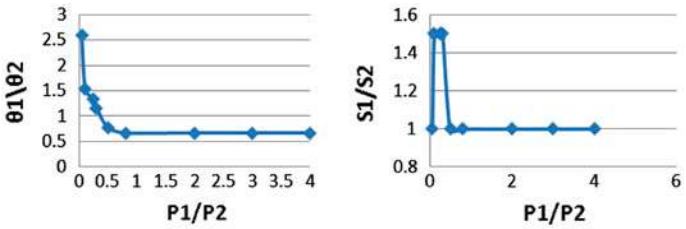


Figure 6. Effect of the variation of the ratio of two products on ratio of completion rates and buffer sizes.

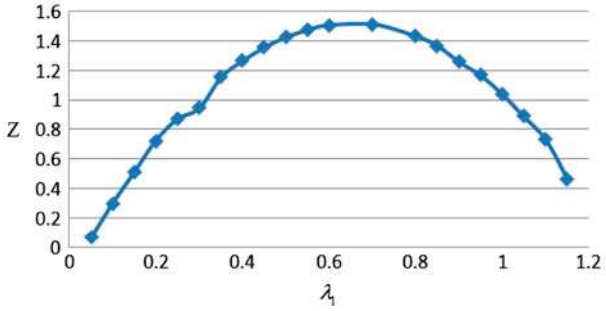


Figure 7. Effect of demand on profit function.

It should be mentioned that only the results of product *I* are disclosed here although the behaviour of product *II* is similar.

- Effect of prices ratio on completion rates ratio and buffer sizes ratio

Figure 6 illustrates how the ratio of the completion rates and buffer size is decreasing in the ratio of prices. Based on these results, whenever the price of product *I* is chosen to be larger than the product *II*, the supply chain is inclined to produce a lower number of product *I* with a lower completion percentage as a result of the variety of customers whom the system serves in virtue of larger demand.

- Effect of demand on profit function

As shown in Figure 7, an increase in average rate of demand, despite an initial increase in profit, follows a decrease due to the augmentation in costs, especially customer waiting costs.

- Sensitivity analysis of the parameters

In Table 5, the cost parameters of each product and the optimal measures for the percent of completion for product of type i in first production stage (θ_i), optimal storage capacity of type i semi-finished products (S_i), and price quoted to product of type i(P_i) are demonstrated.

Table 5. List of cost-related parameters of two products.

		Product *I*				Product *II*
C_{Ui}		0.0001				0.001
C_{Hi}		0.00001				0.001
C_{Wi}		0.01				0.01
C_{Ci}		0.000005				0.0005
$Z\ (P_1, P_2, \theta_1, \theta_2, S_1, S_2)$	P_1	P_2	θ_1	θ_2	S_1	S_2
1.514182834	4	13	0.14	0.11	3	2

Since C_{U_i}, the cost of disposing of an unsuitable item of type i, is lower for product I, the supply chain is prone to produce product I with higher θ. Moreover, the lower cost of C_{W_i}, the cost of customer order fulfilment delay for product of type i, for product I leads to the same result, since a higher completion percentage reduces the time a customer has to wait for the production of his requested product. On the other hand, higher C_{H_i}, the holding cost for semi-finished products of type i, for product II in comparison with product I is conducive to lower buffer size for product II. In addition, lower C_{C_i}, the cost of establishing type i semi-finished products storage capacity, for product I enhances this effect.

6. Conclusion

For the first time, an integrated operations-marketing queuing-based model for multi-product supply chain is developed to help understand how the OPP affects the trade-off between customer order fulfilment delay and inventory risks, when both stages of production take non-negligible time and when the production capacity is limited. In order to evaluate performance measures, a queuing model and the matrix geometric method were applied. In addition, the problem under shared and unshared capacity is developed. We proposed the theoretical model in generic terms and solved the numerical example for a two products supply chain. Our observations, based on extensive numerical experiments, indicate that adding the price to a manufacturing model helps us not only to investigate the effect of price on manufacturing performance indices, but also to establish marketing strategies to increase the profit.

In this paper, the authors seek to develop a model which simultaneously considers the product pricing decision and OPP positioning under uncertain demand and delivery lead-time with price sensitive demand function. Moreover, there is a practical base; integrating operations-marketing perspective by adding decision on product pricing with assumption of price sensitive demand function and theoretical base; applying a queuing approach for modelling the problem because of uncertain nature of demand arrival and lead-time. Finally, this model helps strategic management of SCM to have integrated operations-marketing perspective. Hence, top managers can have a wider view in their decision makings. Following issues can be considered as future research possibilities:

- Applying the capacity constraint in customers queue: This study investigates the simplest model for the queuing systems. It is more realistic, however, to examine other queuing system models such as M/M/m, M/M/m/k, and so forth.
- Relaxing the assumptions of exponentially distributed arrival and service times: The assumption of exponentially distributed arrival and service time can be relaxed by use of G/G/m models.
- Considering the impatient customers in arrival demands.
- Applying other solution methods: It is possible to use other heuristic and meta-heuristic solution method after carefully scrutinising the dimensions of the mathematical model and its attributes.

Acknowledgment

We wish to thank anonymous reviewers for helpful comments and suggestions.

References

Adan, I. J. B. F., and J. Van der Wal. 1998. “Combining Make to Order and Make to Stock.” *OR Spektrum* 20 (2): 73–81.

Ahmadi, M., and E. Teimouri. 2008. “Determining the Order Penetration Point in Auto Export Supply Chain by the Use of Dynamic Programming.” *Journal of Applied Sciences* 8 (18): 3214–3220.

Akkerman, R., D. Van der Meer, and D. P. Van Donk. 2010. “Make to Stock and Mix to Order: Choosing Intermediate Products in the Food-Processing Industry.” *International Journal of Production Research* 48 (12): 3475–3492.

Arreola-Risa, A., and G. A. DeCroix. 1998. “Make-to-Order versus Make-to-Stock in a Production–Inventory System with General Production Times.” *IIE Transactions* 30 (8): 705–713.

Aviv, Y., and A. Federgruen. 2001. “Design for Postponement: A Comprehensive Characterization of Its Benefits under Unknown Demand Distributions.” *Operations Research* 49 (4): 578–598.

Banerjee, A., B. Sarkar, and S. Mukhopadhyay. 2012. “Multiple Decoupling Point Paradigms in a Global Supply Chain Syndrome: A Relational Analysis.” *International Journal of Production Research* 50 (11): 3051–3065.

Bolch, G., S. Greiner, H. de Meer, and S. Trivedi. 1998. *Queuing Networks and Markov Chains: Modeling and Performance Evaluation with Computer Science Applications*. New York: John Wiley.

Boyaci, T., and S. Ray. 2007. *Product Differentiation and Capacity Cost Interaction in Time and Price Sensitive Markets* 5 (1): 18–36.

Carr, S., and I. Duenyas. 2000. "Optimal Admission Control and Sequencing in a Make-to-Stock/Make-to-Order Production System." *Operations Research* 48 (5): 709–720.

Chang, S. H., P. F. Pai, K. J. Yuan, B. C. Wang, and R. K. Li. 2003. "Heuristic PAC Model for Hybrid MTO and MTS Production Environment." *International Journal of Production Economics* 85 (3): 347–358.

Chayet, S., W. Hopp, and X. Xu. 2004. "The Marketing-operations Interface." In *Handbook of Quantitative Supply Chain Analysis: Modeling in the E-business Era*, Springer, Issue Part III. *Supply Chain Coordinations in E-Business* 8.

Choi, K., R. Narasimhan, and S. W. Kim. 2012. "Postponement Strategy for International Transfer of Products in a Global Supply Chain: A System Dynamics Examination." *Journal of Operations Management* 30 (3): 167–179.

Erickson, G. M. 2011. "A Differential Game Model of the Marketing-Operations Interface." *European Journal of Operational Research* 211 (2): 394–402.

Feng, Y., S. D'Amours, and R. Beauregard. 2008. "The Value of Sales and Operations Planning in Oriented Strand Board Industry with Make-to-Order Manufacturing System: Cross Functional Integration under Deterministic Demand and Spot Market Recourse." *International Journal of Production Economics* 115 (1): 189–209.

Feng, Y., S. D'Amours, and R. Beauregard. 2010. "Simulation and Performance Evaluation of Partially and Fully Integrated Sales and Operations Planning." *International Journal of Production Research* 48 (19): 5859–5883.

Günalay, Y. 2011. "Efficient Management of Production-Inventory System in a Multi-Item Manufacturing Facility: MTS vs. MTO." *The International Journal of Advanced Manufacturing Technology* 54 (9–12): 1179–1186.

Gupta, D., and S. Benjaafar. 2004. "Make-to-Order, Make-to-Stock, or Delay Product Differentiation? A Common Framework for Modeling and Analysis" *IIE Transactions* 36 (6): 529–546.

Hallgren, M., and J. Olhager. 2006. "Differentiating Manufacturing Focus." *International Journal of Production Research* 44 (18–19): 3863–3878.

Harrison, A., and H. Skipworth. 2008. "Implications of Form Postponement to Manufacturing: A Cross Case Comparison." *International Journal of Production Research* 46 (1): 173–195.

Ho, T. H., and Y. S. Zheng. 2004. "Setting Customer Expectation in Service Delivery: An Integrated Marketing-Operations Perspective." *Management Science* 50 (4): 479–488.

Hoekstra, S., J. Romme, and S. Argelo. 1992. *Integral Logistic Structures: Developing Customer-Oriented Goods Flow*. New York: Industrial Press.

Ioannidis, S., and V. Kouikoglou. 2008. "Revenue Management in Single-Stage CONWIP Production Systems." *International Journal of Production Research* 46 (22): 6513–6532.

Jeong, I. J. 2011. "A Dynamic Model for the Optimization of Decoupling Point and Production Planning in a Supply Chain." *International Journal of Production Economics* 131 (2): 561–567.

Jewkes, E. M., and A. S. Alfa. 2009. "A Queuing Model of Delayed Product Differentiation." *European Journal of Operational Research* 199 (3): 734–743.

Kerkkänen, A. 2007. "Determining Semi-Finished Products to Be Stocked When Changing the MTS-MTO Policy: Case of a Steel Mill." *International Journal of Production Economics* 108 (1–2): 111–118.

Kumar, S., D. A. Nottestad, and J. F. Macklin. 2007. "A Profit and Loss Analysis for Make-to Order versus Make-to-Stock Policy: A Supply Chain Case Study." *Engineering Economist* 52 (2): 141–156.

Lee, H. L., and C. S. Tang. 1997. "Modelling the Costs and Benefits of Delayed Product Differentiation." *Management Science* 43 (1): 40–53.

Mikkola, J. H., and T. Skjøtt-Larsen. 2004. "Supply-Chain Integration: Implications for Mass Customization, Modularization and Postponement Strategies." *Production Planning & Control* 15 (4): 352–361.

Neuts, M. F. 1981. *Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach*. Mineola, NY: Dover Pubns.

O'Leary-Kelly, S. W., and B. E. Flores. 2002. "The Integration of Manufacturing and Marketing/Sales Decisions: Impact on Organizational Performance." *Journal of Operations Management* 20 (3): 221–240.

Olhager, J. 2003. "Strategic Positioning of the Order Penetration Point." *International Journal of Production Economics* 85 (3): 319–329.

Olhager, J. 2010. "The Role of the Customer Order Decoupling Point in Production and Supply Chain Management." *Computers in Industry* 61 (9): 863–868.

Oliva, R., and N. Watson. 2011. "Cross-Functional Alignment in Supply Chain Planning: A Case Study of Sales and Operations Planning." *Journal of Operations Management* 29 (5): 434–448.

Perona, M., N. Saccani, and S. Zanoni. 2009. "Combining Make-to-Order and Make-to-Stock Inventory Policies: An Empirical Application to a Manufacturing SME." *Production Planning & Control* 20 (7): 559–575. doi:10.1080/09537280903034271.

Quante, R., H. Meyr, and M. Fleischmann. 2009. "Revenue Management and Demand Fulfillment: Matching Applications, Models, and Software." *OR Spectrum* 31 (1): 31–62. doi:10.1007/s00291-008-0125-8.

Rafiei, H., and M. Rabbani. 2012. "Capacity Coordination in Hybrid Make-to-Stock/Make-to-Order Production Environments." *International Journal of Production Research* 50 (3): 773–789.

Rajagopalan, S. 2002. "Make-to-Order or Make-to-Stock: Model and Application." *Management Science* 48 (2): 241–256.

Rao, V. R. 2009. *Handbook of Pricing Research in Marketing*. Cheltenham, UK: Edward Elgar Pub.

Ray, S. 2005. "An Integrated Operations–Marketing Model for Innovative Products and Services." *International Journal of Production Economics* 95 (3): 327–345.

Rudberg, M., and J. Wikner. 2004. "Mass Customization in Terms of the Customer Order Decoupling Point." *Production Planning & Control* 15 (4): 445–458.

Shao, X. F., and M. Dong. 2012. "Comparison of Order-Fulfilment Performance in MTO and MTS Systems with an Inventory Cost Budget Constraint." *International Journal of Production Research* 50 (7): 1917–1931.

Sharman, G. 1984. "The Rediscovery of Logistics." *Harvard Business Review* 62 (5): 71–79.

Skipworth, H., and A. Harrison. 2004. "Implications of Form Postponement to Manufacturing: A Case Study." *International Journal of Production Research* 42 (10): 2063–2081.

Skipworth, H., and A. Harrison. 2006. "Implications of Form Postponement to Manufacturing a Customized Product." *International Journal of Production Research* 44 (8): 1627–1652.

Soman, C. A., D. P. Van Donk, and G. Gaalman. 2004. "Combined Make-to-Order and Make-to-Stock in a Food Production System." *International Journal of Production Economics* 90 (2): 223–235.

Sox, C. R., L. J. Thomas, and J. O. McClain. 1997. "Coordinating Production and Inventory to Improve Service." *Management Science* 43 (9): 1189–1197.

Su, J. C. P., Y. L. Chang, M. Ferguson, and J. C. Ho. 2010. "The Impact of Delayed Differentiation in Make-to-Order Environments." *International Journal of Production Research* 48 (19): 5809–5829.

Sun, X., P. Ji, L. Sun, and Y. Wang. 2008. "Positioning Multiple Decoupling Points in a Supply Network." *International Journal of Production Economics* 113 (2): 943–956.

Tang, C. S. 2010. "A Review of Marketing-Operations Interface Models: From Co-Existence to Coordination and Collaboration." *International Journal of Production Economics* 125 (1): 22–40.

Teimoury, E., and M. Fathi. 2012. "A Queuing Approach for Making Decisions About Order Penetration Point in Supply Chain with Impatient Customer." *The International Journal of Advanced Manufacturing Technology* 63 (1–4): 359–371.

Teimoury, E., M. Modarres, F. Ghasemzadeh, and M. Fathi. 2010. "A Queuing Approach to Production-Inventory Planning for Supply Chain with Uncertain Demands: Case Study of PAKSHOO Chemicals Company." *Journal of Manufacturing Systems* 29 (2–3): 55–62.

Teimoury, E., M. Modarres, A. K. Monfared, and M. Fathi. 2011. "Price, Delivery Time, and Capacity Decisions in an M/M/1 Make-to-Order/Service System with Segmented Market." *The International Journal of Advanced Manufacturing Technology* 57 (1–4): 235–244.

Teimoury, E., M. Modarres, I. Khondabi, and M. Fathi. 2012. "A Queuing Approach for Making Decisions about Order Penetration Point in Multiechelon Supply Chains." *The International Journal of Advanced Manufacturing Technology.* doi:10.1007/s00170-012-3913-x.

Tsay, A. A., and N. Agrawal. 2000. "Channel Dynamics under Price and Service Competition." *Manufacturing & Service Operations Management* 2 (4): 372–391.

Van Donk, D. P. 2001. "Make to Stock or Make to Order: The Decoupling Point in the Food Processing Industries." *International Journal of Production Economics* 69 (3): 297–306.

Vandaele, N., and L. Perdu. 2010. "The Operations-finance Interface: An Example from Lot Sizing." Paper presented at the 7th International Conference on Service Systems and Service Management (ICSSSM).

Vidyarthi, N., S. Elhedhli, and E. Jewkes. 2009. "Response Time Reduction in Make-to-Order and Assemble-to-Order Supply Chain Design." *IIE Transactions* 41 (5): 448–466.

Wang, F., R. Piplani, Y. Roland, and E. Lee. 2011. "Development of an Optimal Decision Policy for MTS-MTO System." Paper presented at the POM 22nd Annual Conference, Reno, Nevada, USA.

Wee, K., and M. Dada. 2010. "A Make-to-Stock Manufacturing System with Component Commonality: A Queuing Approach." *IIE Transactions* 42 (6): 435–453.

Wikner, J., and M. Rudberg. 2005. "Introducing a Customer Order Decoupling Zone in Logistics Decision-Making." *International Journal of Logistics: Research and Applications* 8 (3): 211–224.

Wong, H., and D. Eyers. 2011. "An Analytical Framework for Evaluating the Value of Enhanced Customisation: An Integrated Operations-Marketing Perspective." *International Journal of Production Research* 49 (19): 5779–5800.

Wong, H., J. Wikner, and M. Naim. 2009. "Analysis of Form Postponement Based on Optimal Positioning of the Differentiation Point and Stocking Decisions." *International Journal of Production Research* 47 (5): 1201–1224.

Wong, H., J. Wikner, and M. Naim. 2010. "Evaluation of Postponement in Manufacturing Systems with Non-Negligible Changeover times." *Production Planning & Control* 21 (3): 258–273.

Yáñez, F. C., J. M. Frayret, F. Léger, and A. Rousseau. 2009. "Agent-Based Simulation and Analysis of Demand-Driven Production Strategies in the Timber Industry." *International Journal of Production Research* 47 (22): 6295–6319.

Yang, B., and N. Burns. 2003. "Implications of Postponement for the Supply Chain." *International Journal of Production Research* 41 (9): 2075–2090.

Yang, B., N. D. Burns, and C. J. Backhouse. 2004. "Postponement: A Review and an Integrated Framework." *International Journal of Operations & Production Management* 24 (5): 468–487.

Appendix A

$$D_i = \begin{bmatrix} D_{i_{0,0}} & D_{i_{0,1}} & & & \\ & D_{i_{1,1}} & D_{i_{1,2}} & & \\ & & \ddots & \ddots & \\ & & D_{i_{S_i-1,S_i-1}} & D_{i_{S_i-1,S_i}} & \\ & & & D_{i_{S_i,S_i}} & \end{bmatrix}_{(S_i+1)\times(S_i+1)} \tag{A.1}$$

$$D_{i_{m,m}} = \begin{cases} -(\lambda_i + \frac{\mu_i(1-\phi_i)}{\theta_i}) & 1 \le i \le L, \quad 0 \le m \le S_i - 1 \\ -\lambda_i & 1 \le i \le L, \quad m = S_i \end{cases}$$

$$D_{i_{m,m+1}} = \frac{\mu_i(1-\phi_i)}{\theta_i} \qquad 1 \le i \le L,\ 0 \le m \le S_i - 1$$

$$E_i = \begin{bmatrix} E_{i_{0,0}} & E_{i_{0,1}} & & & \\ & E_{i_{1,1}} & E_{i_{1,2}} & & \\ & & \ddots & \ddots & \\ & & E_{i_{S_i-1,S_i-1}} & E_{i_{S_i-1,S_i}} & \\ & & & E_{i_{S_i,S_i}} & \end{bmatrix}_{(S_i+1)\times(S_i+1)}$$

$$E_{i_{m,m}} = \begin{cases} -\left(\lambda_i + \dfrac{\mu_i(1-\phi_i)}{\theta_i}\right) & 1 \le i \le L,\ m = 0 \\ -\left(\lambda_i + \dfrac{\mu_i(1-\phi_i)}{\theta_i} + \dfrac{\mu_i}{1-\theta_i}\right) & 1 \le i \le L,\ 1 \le m \le S_i - 1 \\ -\left(\lambda_i + \dfrac{\mu_i}{1-\theta_i}\right) & 1 \le i \le L,\ m = S_i \end{cases} \tag{A.2}$$

$$E_{i_{m,m+1}} = \frac{\mu_i(1-\phi_i)}{\theta_i} \qquad 1 \le i \le L,\ 0 \le m \le S_i - 1$$

$$C_i = \begin{bmatrix} 0 & \mathbf{0} \\ I\dfrac{\mu_i}{1-\theta_i} & \mathbf{0} \end{bmatrix}_{(S_i+1)\times(S_i+1)} \tag{A.3}$$

$$A_i = [I\lambda_i]_{(S_i+1)\times(S_i+1)} \tag{A.4}$$

Appendix B

$$F_{i_{m,m}} = \begin{cases} -\left(\dfrac{\mu_i(1-\phi_i)}{\theta_i}\right) & 1 \le i \le L,\ m = 0 \\ -\left(\dfrac{\mu_i(1-\phi_i)}{\theta_i} + \dfrac{\mu_i}{1-\theta_i}\right) & 1 \le i \le L,\ 1 \le m \le S_i - 1 \\ -\left(\dfrac{\mu_i}{1-\theta_i}\right) & 1 \le i \le L,\ m = S_i \end{cases} \tag{B.1}$$

$$F_{i_{m,m+1}} = \frac{\mu_i(1-\phi_i)}{\theta_i} \qquad 1 \le i \le L,\ 0 \le m \le S_i - 1 \tag{B.2}$$

$$F_{i_{m,m-1}} = \frac{\mu_i}{1-\theta_i} \qquad 1 \le i \le L,\ 1 \le m \le S_i \tag{B.3}$$

Journal of Manufacturing Systems 37 (2015) 265–276

Contents lists available at ScienceDirect

Journal of Manufacturing Systems

journal homepage: www.elsevier.com/locate/jmansys

Modeling the merging capacity for two streams of product returns in remanufacturing systems

Mahdi Fathi [a], Farshid Zandi [b], Oualid Jouini [a,*]

[a] Ecole Centrale Paris, Laboratoire Génie Industriel, Grande Voie des Vignes, 92290 Châtenay-Malabry, France
[b] Department of Industrial Engineering, K.N Toosi University of Technology, Tehran, Iran

ARTICLE INFO

Article history:
Received 16 January 2014
Received in revised form 19 July 2014
Accepted 29 August 2014
Available online 27 September 2014

Keywords:
Remanufacturing
Closed loop supply chain
Queueing systems
Admission decision problem
Time value

ABSTRACT

We consider a remanufacturing system with two streams of returned products and different variability levels (high and low). The arrival of returns with high variability is modeled with a hyperexponential renewal process and that of returns with low variability is modeled with a Poisson process. The remanufacturing facility can process the returned products in two ways. For the first way, each type of returns is remanufactured by a dedicated capacity. For the second way, returns from two different markets are remanufactured by a merged capacity.

Analytical queueing models with the time value of money consideration are proposed for the admission decision, which decides on the acceptance or not of returned products based on quality and processing time. The proposed modeling determines the admission decision threshold value in order to maximize the total expected profit of the remanufacturing system. Our analysis also allows to study the interaction between the overall utilization and the arrival process variability. The results show the impact of the model parameters on the admission decision value and the total expected discounted profit. Also, the total expected discounted profit under the separated and merged capacities are compared.

1. Introduction

Product remanufacturing has been developed rapidly aiming to protect the environment and to reduce production costs in the supply chain. In today's market, consumers are usually allowed to return a purchased product. Many returned products are sometimes remanufactured and reused without even the customer knowledge [43]. Due to the large amount of returned products, the manufacturers should consider these returns in the production planning and inventory control processes. This is a new important issue for manufacturing systems. Remanufacturing is defined by [46] as: "An industrial process in which worn out products are restored to seem like new ones". Consider a capacitated facility which remanufactures returns to remarket as remanufactured products. High congestion levels at the remanufacturing facility may cause considerable delays and consequently remarketing value losses for time-sensitive products. By the development of technology, especially among electronic products, the useful lives of products are shortened. Making decision on return of products plays an important role for high remanufacturing costs and short product life cycles. Remanufacturing all returned products might not be possible because of increasing remanufacturing costs according to the spent time. Admission decision for remanufacturing is based on the quality and the required processing times of returned products. These are the main source of uncertainty. The returned products can be then classified into: waste to be disposed, or material and parts to be used in processes for producing parts and products. One possibility for a heavily-loaded remanufacturing facility, when the queue at the remanufacturing facility becomes too long, is to sell returned products as-is immediately at their salvage value.

Research on remanufacturing systems has been done under various perspectives. Remanufacturing is an important activity in closed loop supply chains (CLSC). Hence it has been successfully practiced in many industries, such as mobile phones, computers, cameras, and photocopiers. Guide and van Wassenhove [28] further investigated a closed loop supply chain where the quantity, quality, and timing of returns can be controlled by the price offered to buy back the used products. The demand and return rates are assumed to be price-sensitive. Inderfurth [33] investigated the impact of uncertainties on recovery behavior in a closed loop system. Through a numerical analysis, it is shown that the

* Corresponding author. Tel.: +33 141131502.
E-mail addresses: mahdi.fathi@ecp.fr (M. Fathi), f.zandi@sina.kntu.ac.ir (F. Zandi), oualid.jouini@ecp.fr (O. Jouini).

http://dx.doi.org/10.1016/j.jmsy.2014.08.006
0278-6125/

product recovery management becomes much difficult, as the manufacturer should balance the production, recovery, and disposal decisions under considerable uncertainties of demand and return. New integrated models still need to be developed to link various disciplinary perspectives of CLSC [29] and the stochastic nature of demand and return should be paid more attention [50]. For other various quantitative studies on CLSC activities, we also refer the reader to Dekker et al. [13] and Shi et al. [51].

The motivation for this study is that remanufacturing systems are showing an increasing interest in incorporating the merging as an important input into the closed loop supply chains. Moreover, high congestion levels for returns at remanufacturing facility causes substantial delays and consequently remarketing value losses for time-sensitive products and high-tech products with short life cycles, such as consumer electronic equipment computers and printers. Guide et al. [26] report that prices of printers decay at 1% per week. Some PC components decay at even higher rates: 15% per month for compact flash memories and 8% per month for disk drives. Also, several recent trends motivate companies to merge the capacities that were previously dedicated to dissimilar demand processes. There are real case studies of such dissimilarity in demand processes such as: the case of Volvo heavy truck division distribution center that was studied by Narus and Anderson [48], the merging production capacity Alcan Aluminum Ltd. and Arco's Atlantic Ritchfield & Co. that was studied by Iyer and Jain [36]. Therefore, we believe that the merging perspective is needed to determine the admission decision threshold value that decides about acceptance of returned products to prevent the value losses for time-sensitive products.

In this paper, we focus on a remanufacturing system for a type of short life cycle product with stochastic serviceable demand and stochastic returns. There are two return streams with different variabilities in the process of arrivals, namely we consider a hyper-exponential renewal process and a Poisson process. We use an economic framework and the M/M/1, H/M/1, and H_2M/M/1 queues to model the considered remanufacturing processes. We determine the admission decision threshold value that decides about the acceptance of returned products on the base of quality and processing time while maximizing the total expected profit. We show that the difference in variability in arrivals has a significant impact on the value of merging capacity. The proposed modeling aims to address the question: When does merging generate Pareto-improving benefits over the separated system?

The reminder of this paper is organized as follows. In Section 2, we survey the literature related to this paper. In Section 3, we give the description of the remanufacturing system under consideration. Section 4 is devoted to the problem formulation and the theoretical analysis of the queueing modeling. In Section 5, we conduct a numerical study to illustrate the theoretical results. The paper ends with concluding remarks and directions for future research.

2. Literature review

In the literature, hybrid production processes are modeled using capacitated and incapacitated models. The capacitated and incapacitated models both in manufacturing and remanufacturing processes are modeled as queuing networks with finite production rates [1,44,57,25].

Ching et al. [11] studied a Markovian queueing modeling for hybrid manufacturing/remanufacturing systems. They assumed that the arrival of returns follows a Poisson process and there is not any rejection of returns from the system. A matrix geometric method is applied to analyze the resulting queueing network. Inderfurth and van der Laan [34] studied a remanufacturing system and proposed a model where demands from customers can be satisfied by both new and recovered products. The recovered products were disposed or stocked in a dedicated inventory. Mahadevan et al. [47] used a similar modeling and proposed pull and push inventory policy for the remanufacturing system. Kiesmüller and van der Laan [42] considered dependent returned products and customer demands in the remanufacturing system. Karamouzian et al. [40] provided an analytical queueing analysis to obtain the best policy to accept returned products. Furthermore, a continuous genetic algorithm is implemented to solve the model, which happens to be a mixed integer non-linear mathematical program.

There is a rich literature that investigated production planning and control for remanufacturing, but only a few of these studies considered the quality of returned products. Returns are often assumed to have one single quality level [56,57,55,21,58,16]. Souza et al. [52] modeled the remanufacturing facility as a multiclass open queueing network where quality levels of returned products determine their classes. They dedicated special remanufacturing stations for different quality type returns. They examined the dispatching rules in remanufacturing stations in order to reduce flow times and improve the service level. Galbreth and Blackburn [19] considered a remanufacturing system with both deterministic demand and random demand under used product variability condition. In order to analyze remanufacturing and disposal decision, Aras et al. [4] emphasized on quality levels of returned product and constructed a continuous time Markov chain model and investigated quality based remanufacturing lead times and disposal cost. Takahashi et al. [53] used Markov analysis to study a remanufacturing system where recovered products are decomposed and classified into wasted to be disposed and materials and parts to be used in the processes for producing parts and products. Recently, Jin et al. [38] investigated the assembly strategies for product remanufacturing with variation in the quality level of returns. The author studied the optimal policy for the modular product reassembly within a remanufacturing setting where a firm receives returns with different quality levels and reassembles products of multiple classes to customer orders. Moreover, [39] modeled performance analysis of a remanufacturing system with warranty return admission.

Behret and Korugan [8] analyzed a hybrid manufacturing/remanufacturing system under general distributed processing times with different variances. Behret and Korugan [9]used simulation to analyze a hybrid system under uncertainties in the quality of remanufactured products, return rate and return times. Dobos and Richter [14] studied the quality of used products in an integrated production recycling system, and showed that it is better for the manufacturer to only buy back reusable products. Also, many applications and methods for analyzing the hybrid manufacturing system are discussed in [5–7,60,1,41].

Numerous recent trends motivate companies to merge their capacities which were previously dedicated to dissimilar demand processes. There are real case studies of such dissimilarity in demand processes. Gupta and Gerchak [30] provided several examples on the issue of operational synergies in a merger/acquisition between parties with different characteristics. Narus and Anderson [48] studied the case of Volvo heavy truck division distribution center which has separate distribution capacities to serve urgent and scheduled orders. It should be noted that the urgent orders have more variability than the scheduled ones. Eisenstein and Iyer [15] discussed the Chicago school system in which two separated distribution capacities and warehouses were used to serve demands with different levels of predict ability. Fisher [17] investigated two product types with different variabilities in demand: functional and innovative. The demand processes of functional products are less variable than the innovative products. Lee and Tang [45] studied modularization and part commonality term in manufacturing

systems and suggested that the redesigned parts can be produced at the same manufacturing capacity. Consequently, the separate production processes for parts can be removed. Jain [37] considered the value of merging in supply chains which serve product demands with different variability and analyzed models which integrate production queueing models with base stock inventory systems serving demands with different inter-arrival time distributions. Recently, Flapper et al. [18] used Markov decision processes to analyze a hybrid manufacturing–remanufacturing system in which demand and used products arrive via mutually independent Poisson processes. Manufacturing and remanufacturing operations are executed by a single shared resource.

There are few researches in which economic aspect of remanufacturing systems is considered. Geyer et al. [20] studied the economics of remanufacturing under limited component durability and finite product life cycles. Also, Guide et al. [24] proposed a two-step heuristic policy for a busy remanufacturing facility. In a first step, the returned product's random processing time is observed and in a second step a disposition decision is made: if the processing time is larger than a specified value, the product is salvaged; otherwise, the product is remanufactured. Harrison [32] considered the dynamic scheduling problem for the multi-class single server queueing system with discounted rewards. Guide et al. [24] applied Harrison's result to the problem with positive salvage value.

Queueing theory has been extensively adopted to analyze a variety of performance analysis problems in manufacturing systems [22,3]. Queueing models, in turn, can be categorized as descriptive (provide values for performance measures of interest for a given configuration) or prescriptive (provide guidelines for running the system most effectively). Fu [22] conducted a comprehensive survey on queueing models for manufacturing applications. Markovian queueing models lead to a useful tool for modeling manufacturing systems in general [10] and flexible manufacturing systems in particular [12].

The above literature does not consider the impact of delays in the remanufacturing facility and does not consider the decay rate in price for time-sensitive products and high-tech products with short life cycles. In this paper, we provide an analytical queueing model for the optimal disposition decision for product returns in a remanufacturing system. It is assumed that a returned product is first tested and would consequently be remanufactured. The proposed model tries to find the optimal disposition decision for product returns by providing an approximate procedure to compute the optimal threshold value on the processing time. It also considers the quality of returned products for remanufacturing. We moreover examine whether merged remanufacturing capacity can increase the revenue at each returned product stream and determine a Pareto improving region. Our proposed modeling differs from the studied papers in several ways. First, the returned products come from two different markets with different arrival variabilities. Second, the merging of facility remanufacturing capacity is investigated. Finally, the time value of money and remarketing value losses for time-sensitive products and high-tech products with short life cycles are taken into account.

3. Problem description and notations

Consider a hybrid make-to-stock production system. It consists of two independent manufacturing and remanufacturing processes where the first manufactures new products from raw materials while the second remanufactures returned items. The finished products coming from manufacturing or remanufacturing are stored in a common central warehouse, from which a random customer demand is satisfied. Two demand classes are considered, namely the domestic market (distribution companies, Market H) and the international market (export department, Market M). Fig. 1a illustrates the proposed model. At the remanufacturing system, returned products are first inspected and classified according to their quality and processing time, and are then remanufactured using a single server facility (Fig. 1b).

We assume that the arrival process of returned products from market M have lower variability than that from market H. As commonly used in the literature [61,31], the arrival of returned products from Market M is assumed to follow a Poisson process with mean rate λ_m. The inter-arrival times between two successive returns from Market H is assumed to follow a hyperexponential distribution with mean intensity λ_h and coefficient of variation *hcv*. The choice of the hyperexponential distribution comes from the fact that it has a coefficient of variation higher than 1 (coefficient of variation of an exponential distribution), which allows to make the difference between the variabilities of the two return processes.

The capacitated remanufacturing facility can refurbish the two types of returned products in two manners. A separated way where each type is remanufactured with a dedicated capacity, and a merged way where the returned products from the two types join the same queue and wait to be remanufactured. In the separated system, each returned product stream has a dedicated remanufacturing capacity. The separated remanufacturing system is modeled as an $M/M/1$ queue for market M, and as an $H/M/1$ queue for market H. In the merged system, the two streams of returned products join a single FCFS queue. The merged system is modeled as a $H_2M/M/1$ queue. Through this modeling, a benefit-cost function for the remanufacturing system is presented. The proposed cost function uses a threshold value as a measure to analyze and determine an admission decision for returned products. Further details on the demand modeling and the testing stage are given next in Sections 3.1 and 3.2, respectively. For ease of presentation, we give in what follows the list of the notations used for the analysis in this paper.

Notations:

k_m: threshold value for an external market returned product in the separated system (min),

k_h: threshold value for a domestic market returned product in the separated system (min),

k_m^M: threshold value for an external market returned product in the merged system (min),

k_h^M: threshold value of a domestic market returned product in the merged system (min),

λ_h: rate of product returns from the domestic market (min^{-1}),

λ_m: rate of product returns from the external market (min^{-1}),

μ_h: rate of the single exponential remanufacturing server for a product returned from the domestic market (min^{-1}),

μ_m: rate of the single exponential remanufacturing server for a product returned from the external market (min^{-1}),

μ: rate of the single exponential remanufacturing server (min^{-1}), $\mu = \mu_h + \mu_m$,

ρ_h^M: traffic intensity in the remanufacturing queue due to returned products from the domestic market in the merged system, $\rho_h^M = \lambda_h/\mu$,

ρ_m^M: traffic intensity in the remanufacturing queue due to returned products from the external market in the merged system, $\rho_m^M = \lambda_m/\mu$,

ρ^M: total traffic intensity in the remanufacturing queue in the merged system, $\rho^M = \rho_m^M + \rho_h^M < 1$,

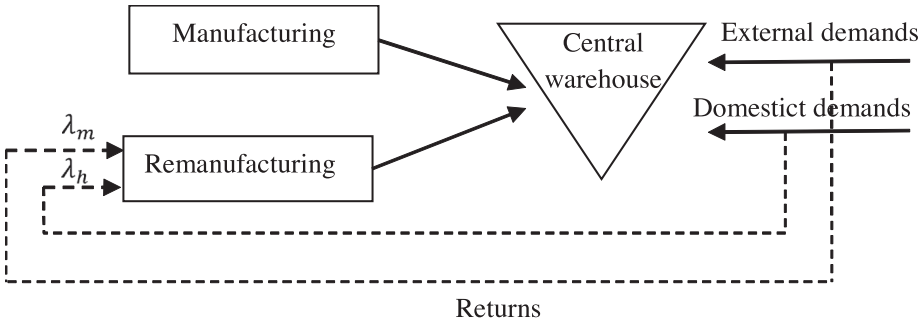
p: salvage value (dollar/unit),

r: obtained revenue from remanufacturing a returned product (dollar/unit),

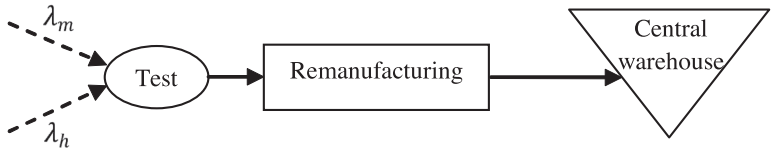
γ: the regular discount rate,

α: the revenue decay rate,

β: the overall discount rate ($\beta = \alpha + \gamma$).



(a) Hybrid make-to-stock production system with central warehouse



(b) Testing stage and remanufacturing model with two return streams

Fig. 1. Framework for hybrid manufacturing/remanufacturing system.

3.1. Demand modeling

The use of the hyperexponential distribution is common in the queueing literature for the modeling of high variability and superposition of high variability arrival processes (see for example [2,59]). It is used to model high variability distributions in the inventory literature and make-to-stock queueing models [49]. Moreover, this modeling can be applied to time varying rate situations, where the corresponding arrival process is a non-homogeneous Poisson process. It should be noted that this arrival process is alternatively on and off for exponential periods. The hyperexponential distribution can be applied for differentiating between the variability of two demand processes based on the single parameter *hcv*.

According to [35], the returned product stream of the domestic market can be modeled as a renewal process where the inter-arrival times follow a hyperexponential distribution of two degrees (H_2). This distribution is a probabilistic mixture of two exponential distributions. It has four parameters k_1, k_2, r_1 and r_2. Its probability density function (pdf), denoted by $a(.)$, is given by

$$a(t) = k_1 r_1 e^{-r_1 t} + k_2 r_2 e^{-r_2 t}, \tag{1}$$

where $0 \le k_1, k_2 \le 1$ *and* $k_1 + k_2 = 1$, and $t \ge 0$. The cumulative distribution function (cdf) of H_2, denoted by $A(.)$, is given by

$$A(t) = 1 - k_1 e^{-r_1 t} - k_2 e^{-r_2 t}, \tag{2}$$

for $t \ge 0$. Applying the Laplace transform on Eq. (2) leads to

$$\zeta(z) = \frac{k_1 r_1}{r_1 + z} + \frac{k_2 r_2}{r_2 + z}, \tag{3}$$

for $z \ge 0$. The balanced mean assumption is primarily used to reduce the number of parameters describing the hyperexponential distribution from three to two; the mean and the coefficient of variation [59]. Using the normalization $k_1/r_1 = k_2/r_2$, the parameters of H_2 are displayed next in Eqs. (4) and (5) [54].

$$k_1 = 0.5\left(1 + \sqrt{\frac{(hcv)^2 - 1}{(hcv)^2 + 1}}\right);\ r_1 = 2k_1\lambda_h, \tag{4}$$

$$k_2 = 1 - k_1, \quad r_2 = 2k_2\lambda_h. \tag{5}$$

Let $c = 2(k_1^2 + k_2^2)$. The parameter c is a measure of variability of the market H demand process where $1 \le c < 2$. By definition, a hyperexponential distribution has $hcv \ge 1$, and this mapping from *hcv* to c allows to compress the entire variability range to $1 \le c < 2$ which would simplify further the analysis. Note that c is increasing in *hcv* and the hyperexponential distribution degenerates into an exponential distribution when $hcv = 1$.

3.2. Testing stage as a class differentiation

In this section, we discuss about the threshold value for the admission decision and also the different classes of returned products to the single testing stage. The results about separated capacity and merged capacity are discussed next in Sections 4.1 and 4.2. The returned products are not exogenous to the remanufacturing problem. Required time and materials to restore the product as a new product by remanufacturing depend on returns condition and quality [27,28]. Returned products from each market are different in terms of their functionalities and types. Hence, incoming returns are different in terms of quality and consequently in terms of processing times. The remanufacturing process time for all returns from different markets is drawn from the same distribution with different rates.

In the considered remanufacturing system, there is an inspection stage with infinite capacity (i.e., with no delay) that determines the required processing time of the returned product. The remanufacturing process time estimated by the testing stage is used to make the admission decision. All the returned products with a processing time greater than a threshold value (denoted by k) are rejected from the remanufacturing process, while those with

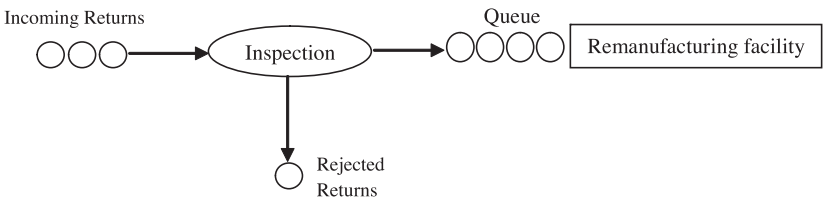


Fig. 2. Admission control of returned products.

an estimated processing time less or equal to k are remanufactured. The threshold value k divides the returned products into two classes: Class 1 returned products comprises products with a remanufacturing time less or equal to k which are admitted to the remanufacturing process. Products with a remanufacturing time higher than k constitute Class 2 returned products that are rejected (recycled). Fig. 2 represents this decision problem in accordance with [24].

4. Problem formulation

This section is devoted to the mathematical formulation of the model. The separated and merged remanufacturing models are analyzed in Sections 4.1 and 4.2, respectively.

4.1. Separated remanufacturing model

In the separated system, each returned stream from a market joins a dedicated remanufacturing queueing system with its own dedicated capacity. As mentioned in Section 3.2, there exist two options at the arrival of a returned product: to admit it in the queue and obtain revenue r at its completion, or to reject it and sell it at a salvage value p. It is assumed that a unit revenue for a completed remanufactured product decreases exponentially with time. According to the remanufacturing system studied by Guide et al. [24] and Harrison [32], the revenue per completed remanufactured product at time t is, $re^{-\beta t}$ where β is the overall discount rate. There are no explicit holding costs. Holding costs are implicitly defined in the overall discount rate β. This method is used for example by Harrison [32] for the optimal static policy that ranks the classes of returned products with salvage value 0. Guide et al. [24] use a similar modeling to that of Harrison in the context of returns from one market.

The remanufacturing processing time for all returned products is a random variable, denoted by X with cdf $F(x)$, for $x \geq 0$. The remanufacturing processing time is estimated by the incapacitated testing stage, and the admission decision is determined through a threshold value on the processing time (Fig. 3).

A returned product is accepted with probability $P(X \leq k_i) = F(k_i)$, for $i \in \{m, h\}$. Therefore, the mean arrival rate of accepted returned products (Class 1) is $\lambda_i P(X \leq k_i) = \lambda_i F(k_i)$, and that of rejected ones (Class 2) is $\lambda_i P(X > k_i) = \lambda_i (1 - F(k_i))$.

The pdf of the remanufacturing processing time for an accepted returned product (Class 1) is $f_1(x) = f(x)/F(k)$, for $0 \leq x \leq k$ and 0 otherwise. In the separated capacity system, the two return streams are remanufactured by their own dedicated remanufacturing capacity. Thus the two queueing systems of interest are $H_2/M/1$ for high variability returns and $M/M/1$ for low variability returns.

We focus on the steady state analysis of the queueing model as shown in Fig. 3. The remanufacturing cost per unit is a function of the observed remanufacturing time x for that unit. Since unit prices for remanufactured products are independent of x, but decay exponentially with time t, net revenue per completed returned product (price minus cost) at time t depends on its remanufacturing time x according to $r(t, x) = r_0(x)e^{-\alpha t}$, where α is the net revenue decay rate parameter. It is important to mention that the proposed model focuses on discount and decay rate of the returned products in the time at which returned products are in the remanufacturing system. The additional assumption is that the time of holding (inventory) and waiting to transport from market to the manufacturing places are equal to zero. The expected flow time of an accepted returned product from Market M through the remanufacturing process is the expected flow time in an $M/M/1$ queue. From [23], it may be written as $W(k_m) = 1/\mu_m - \lambda_m F(k_m)$.

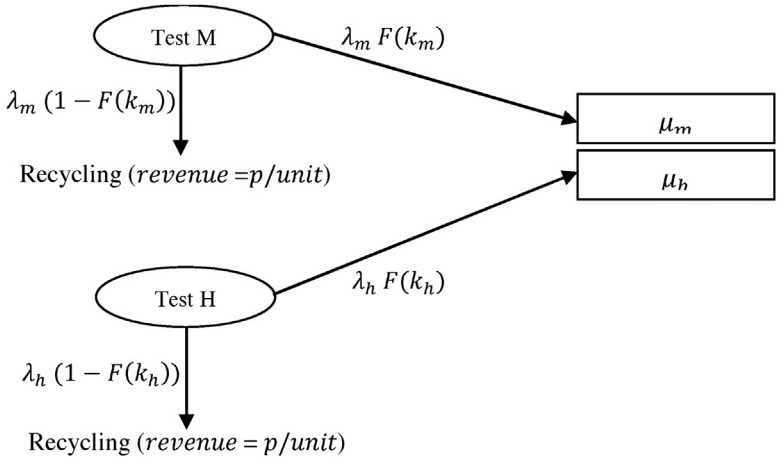


Fig. 3. The separated remanufacturing system.

We denote by V_{mk} the total expected discounted profit with a continuous regular discount rate γ, over an infinite horizon. We have

$$\begin{aligned} V_{mk} &= \lambda_m F(k_m)\int_0^{\infty}\int_0^{k_m} r(t+W(k_m),u) f_1(u) e^{-\gamma(t+W(k_m))}\mathrm{d}u\mathrm{d}t \\ &\quad +\lambda_m\ (1-F(k_m))p\int_0^{\infty} e^{-\gamma t}\mathrm{d}t \\ &= \lambda_m\ F(k_m)\int_0^{\infty}\int_0^{k_m} r_0(u)\frac{f(u)}{F(k)} e^{-\beta(t+W(k_m))}\mathrm{d}u\mathrm{d}t \\ &\quad +\lambda_m(1-F(k_m))\frac{p}{\gamma} \\ &= \frac{\lambda\ e^{-\beta(W(k_m))}}{\beta}\int_0^{k_m} r_0(u) f(u)\mathrm{d}u + \frac{\lambda p(1-F(k_m))}{\gamma}. \end{aligned} \tag{6}$$

The optimal threshold value, k_m^*, can be computed through a numerical method ensuring that the total expected discounted profit is maximized, i.e.,

$$k_m^* = arg\max\ (V_{mk}). \tag{7}$$

The expected flow time of an accepted returned product from Market H through the manufacturing process is the expected flow time in an $H_2/M/1$ queue. Using Iyer and Jain [36] (see the proof of Proposition 3.5, page 1088), we state that

$$W(k_h) = \frac{1}{\mu_h - \mu_h\omega_2^s}, \tag{8}$$

where ω_2^s is the unique root in (0, 1) of the equation $\omega = \zeta[\mu_h(1-\omega)]$. We denote by V_{hk} the total expected discounted profit with a continuous regular discount rate γ, over an infinite horizon. We have

$$\begin{aligned} V_{hk} &= \lambda_h\ F(k_h)\int_0^{\infty}\int_0^{k_h} r(t+W(k_h),u) f_1(u) e^{-\gamma(t+W(k_h))}\mathrm{d}u\mathrm{d}t \\ &\quad +\lambda_h\ (1-F(k_h))p\int_0^{\infty} e^{-\gamma t}\ \mathrm{d}t \\ &= \lambda_h\ F(k_h)\int_0^{\infty}\int_0^{k_h} r_0(u)\frac{f(u)}{F(k_h)} e^{-\beta(t+W(k_h))}\mathrm{d}u\mathrm{d}t \\ &\quad +\lambda_h(1-F(k_h))\frac{p}{\gamma} \\ &= \frac{\lambda e^{-\beta(W(k_h))}}{\beta}\int_0^{k_h} r_0(u) f(u)\mathrm{d}u + \frac{\lambda p(1-F(k_h))}{\gamma}. \end{aligned} \tag{9}$$

The optimal value of the threshold value k_h^* can be again computed through a numerical method ensuring that the total expected discounted profit is maximized.

The random flow time is approximated for each return stream with its expected value $W(k_i)$, for $i \in \{m, h\}$. Therefore the obtained optimal value of the threshold value, k_i^*, is an approximated value. One may use simulation to obtain the exact value of k_i^*. The simulation procedure is however computationally more burdensome while the approximation approach can be implemented in a spreadsheet. More details and an illustration of this approximation is given later in Section 5.

4.2. Merged remanufacturing model

In the merged capacity system, the remanufacturing capacities are combined into a single capacity which is modeled as a single exponential server. The inspection system of each market continues to have separate ownership (revenue and salvage value of each market are independent). The returned products from two markets join a single queue which is served by the merged capacity server. Fig. 4 shows the merged capacity system.

The analysis of the merged system requires the analysis of the $H_2M/M/1$ queue with a FCFS discipline of service. Similarly to the separated system, an economic framework is used. The expected flow time of an accepted returned product from external market through the remanufacturing process is the expected flow time in an $H_2M/M/1$ queue. Using Iyer and Jain [36] (see Lemma 3.2, page 1087), we obtain

$$W(k_m^M) = \frac{(1-u_1)(1-\rho^M u_1)}{\mu\rho_h^M(1-\rho^M)(2-c)u_1^2}, \tag{10}$$

where u_1 is one of the three roots of $\rho_m^M(\rho_m^M + c\rho_h^M)u^3 - ((\rho_m^M)^2 + 2\rho_m^M + 2\rho_m^M\rho_h^M + 2(\rho_h^M)^2 + c\rho_h^M(1-\rho_h^M))u^2 + (1 + 2\rho_m^M + 2\rho_h^M)u - 1 = 0$ (see Appendix A in [36]).

We denote by V_{mk}^m the total expected discounted profit with a continuous regular discount rate γ, over an infinite horizon. Thus

$$\begin{aligned} V_{mk}^M &= \lambda_m F\left(k_m^M\right)\int_0^{\infty}\int_0^{k_m^M} r\left(t+W\left(k_m^M\right),u\right) f_1(u) e^{-\gamma(t+W(k_m^M))}\mathrm{d}u\mathrm{d}t \\ &\quad +\lambda_m\left(1-F\left(k_m^M\right)\right)p\int_0^{\infty} e^{-\gamma t}\mathrm{d}t \\ &= \lambda_m F\left(k_m^M\right)\int_0^{\infty}\int_0^{k} r_0(u)\frac{f(u)}{F(k)} e^{-\beta(t+W(k_m^M))}\mathrm{d}u\mathrm{d}t \\ &\quad +\lambda_m\left(1-F\left(k_m^M\right)\right)\frac{p}{\gamma} \\ &= \frac{\lambda\ e^{-\beta(W(k_m^M))}}{\beta}\int_0^{k} r_0(u)\ \ f(u)\ \ \mathrm{d}u + \frac{\lambda p(1-F\left(k_m^M\right))}{\gamma}. \end{aligned} \tag{11}$$

Also using Iyer and Jain [36], the expected flow time of an accepted returned product from Market M through the remanufacturing process is

$$\begin{aligned} W(k_h^M) &= \frac{\mu(1-\rho_m^M)W(k_m^M) - 1}{\mu\rho_h^M} \\ &= \frac{(1-\rho_m^M)(1-u_1)(1-\rho^M u_1) - \rho_h^M(1-\rho^M)(2-c)u_1^2}{\mu\rho_h^{M2}(1-\rho^M)(2-c)u_1^2}. \end{aligned} \tag{12}$$

We denote by V_{hk}^m the total expected discounted profit with a continuous regular discount rate γ, over an infinite horizon. Therefore

$$\begin{aligned} V_{hk}^M &= \lambda_m\ \ F\left(k_h^M\right)\int_0^{\infty}\int_0^{k_h^M} r\left(t+W\left(k_h^M\right),u\right) f_1(u) e^{-\gamma\left(t+W\left(k_h^M\right)\right)}\mathrm{d}u\mathrm{d}t \\ &\quad +\lambda_h\left(1-F\left(k_h^M\right)\right)p\int_0^{\infty} e^{-\gamma t}\ \ \mathrm{d}t \\ &= \lambda_m\ \ F\left(k_h^M\right)\int_0^{\infty}\int_0^{k_h^m} r_0(u)\frac{f(u)}{F(k)} e^{-\beta\left(t+W\left(k_h^M\right)\right)}\mathrm{d}u\mathrm{d}t \\ &\quad +\lambda_h\left(1-F\left(k_h^M\right)\right)\frac{p}{\gamma} \\ &= \frac{\lambda\ \ e^{-\beta\left(W\left(k_h^M\right)\right)}}{\beta}\int_0^{k_h^M} r_0(u) f(u)\ \ \mathrm{d}u + \frac{\lambda p\left(1-F\left(k_h^M\right)\right)}{\gamma}. \end{aligned} \tag{13}$$

The total expected discounted profit is obtained from Eqs. (6), (9), (11) and (13). Similarly to the previous section, the above analysis leads to k_i^*, for $i \in \{m, h\}$.

5. Numerical study and sensitivity analysis

In this section, we numerically illustrate the analysis of Section 4. Our objective is to gain understanding of the impact of the

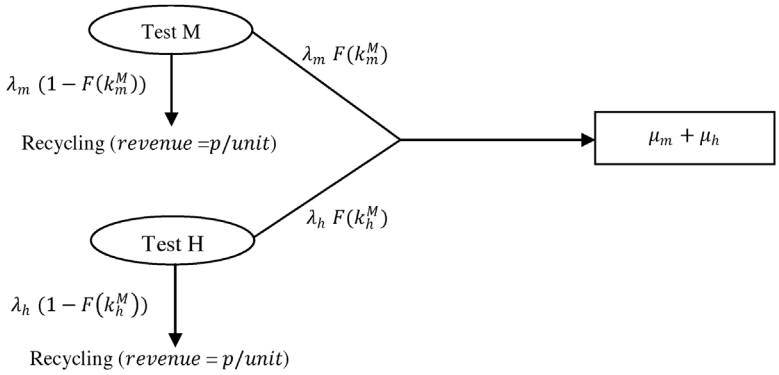


Fig. 4. The merged remanufacturing system.

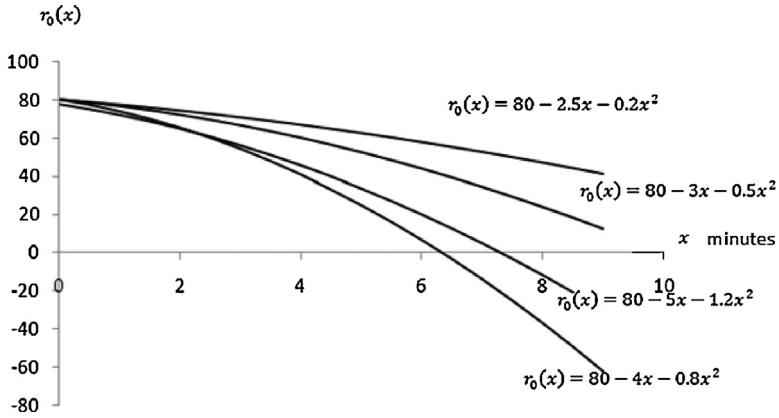


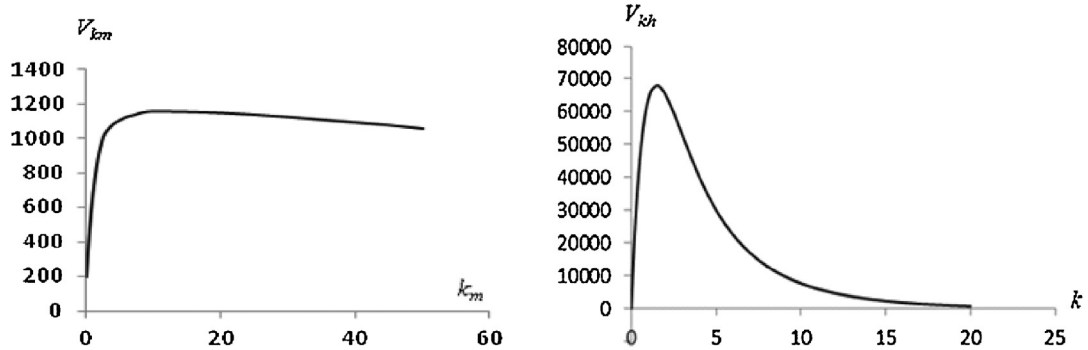
Fig. 5. Remanufacturing cost-time.

model parameters such as returns rate, remanufacturing facility rate, remanufacturing net revenue curve, regular discount rate, revenue decay rate and overall discount rate, on the admission decision and the total expected discounted profit. We also compare between the total expected discounted profits under the two situations of separated and merged capacities.

We moreover examine whether merging remanufacturing capacity will increase the total expected discounted profit which is Pareto-improving. Unit remanufacturing cost for a completed returned product increases with time but profit of this refurbished product is constant. Therefore, the cost-revenue curve can be obtained. According to the real case of Pitney Bowes and Robert Bosch Power Tools (studied by [24]), a quadratic function is used for the net revenue curve $r_0(x) = b_2x^2 + b_1x + b_0$. As shown in Fig. 5, four shapes are considered for the remanufacturing net revenue curve. They represent four different trends of increasing in remanufacturing cost with time. In all curves, b_0 is constant at 80. This is the net margin defined as price minus remanufacturing costs for a product with zero remanufacturing time.

5.1. Analysis of the separated system

In the separated system, there are two separated queueing remanufacturing systems from markets H and M. For each stream of returned product, the threshold value k_i, profit V_i and acceptance percentage $F(k_i)$, for $i \in \{m, h\}$, of a returned product are calculated for different values of returns rate λ_i. We choose the regular discount rate γ, the revenue decay rate α, and the salvage value of external returns p as $\gamma = 0.003$, $\alpha = 0.02$ and, $p = 3$, respectively.



(a) Market M (b) Market H

Fig. 6. The expected profit versus different threshold values k_i ($\gamma = 0.003$, $\alpha = 0.02$ and $p = 3$ and $r_0(x) = 80 - 2.5x - 0.2x^2$).

Table 1
Impact of regular discount and revenue decay rates ($\lambda_m = 0.5$ and $\mu_m = 0.7$).

	γ			
α	0.003	0.005	0.007	0.009
0.01	(11.31,1.00,2725.3)	(11.63,1.00,2338.4)	(11.64,1.00,2042.8)	(11.71,1.00,1809.6)
0.02	(8.02,0.99,1465.8)	(9.15,1.00,1334.8)	(9.44,1.00,1223.6)	(9.47,1.00,1127.9)
0.03	(4.71,0.96,976.5)	(6.38,0.99,908.0)	(7.02,0.99,849.0)	(7.25,1.00,798.3)
0.05	(1.82,0.72,593.6)	(3.16,0.90,535.5)	(3.78,0.93,506.0)	(4.11,0.94,482.2)

Eqs. (6) and (9) are non-linear equations and hard to solve. According to Eq. (7), the optimal value of the threshold k_i^* can be computed using a numerical method that allows to maximize the total expected discounted profit. Fig. 6 provides the plot of Eqs. (6) and (9) versus k_i, for $\mu_i = 0.6$ and $\lambda_i = 0.5$, $i \in \{m, h\}$. Initially we observe an ascending trend in both functions to reach a maximum value and then a descending trend. For the threshold $k_m^* = 10.8$, the maximum of Eq. (6) is equal to 1158.7 and with increasing the threshold value the function gradually decreases. Eq. (9) has the same trend as Eq. (6) and a maximum value equal at 50772.00 for $k_h^* = 0.6$. When the remanufacturing system does not accept any of the returned products ($k_i = 0$) the profit value is negative (small value close to zero) for Market H, while it is positive for Market M ($V_{km} = 200$).

Now, we should discuss about the sensitivity analysis of the threshold value and the expected profit function. The results are shown in Fig. 7a and b with $\mu_m = 0.7$ and $\lambda_m = 0.5$. Fig. 7a-1 reveals that the returned product rate with a constant remanufacturing rate leads to decreasing the threshold value. From Fig. 7a-2, it leads to increasing the expected profit. Fig. 7a-3 reveals that it increases the acceptance percentage. Moreover, Fig. 7b-1, b-2 and b-3 shows that increasing the remanufacturing rate leads to an increase in the threshold value, the expected profit, and the accepted returned.

The results for the sensitivity analysis of α and γ are shown in Table 1. The values of the threshold, the acceptance percentage and the expected total discounted profit for each value of γ and α are written in the table as $(k^*, F(k^*), V(k^*))$. It is observed that for a constant decay rate and an increasing revenue decay rate, the expected discounted profit decreases and the threshold value increases. Therefore, the percentage of accepted returned products decreases. The results illustrate that the lower are the discount and decay rates, the higher is the profit for the remanufacturing system.

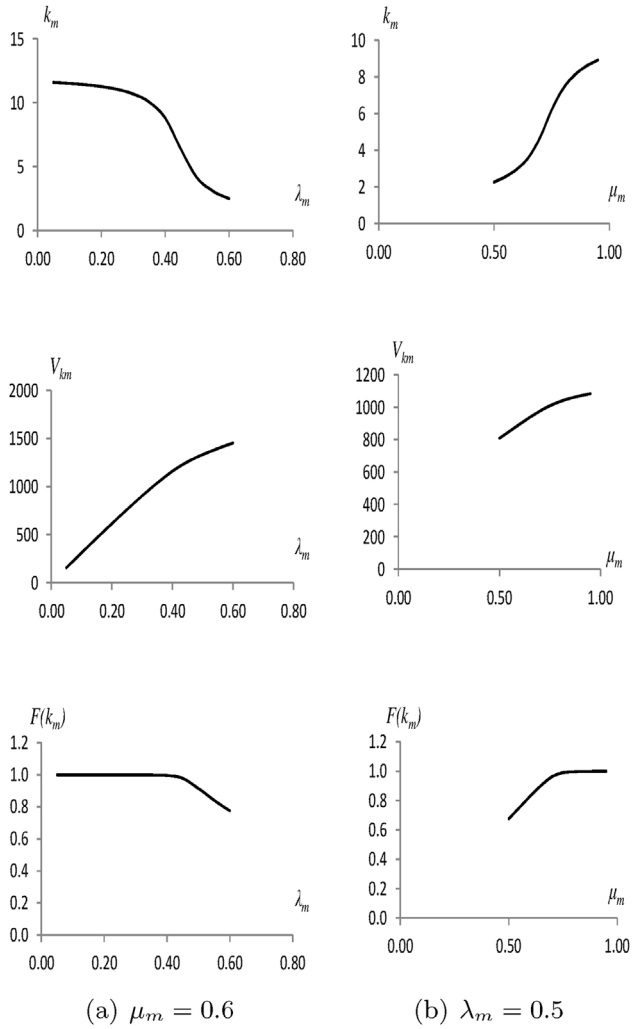


Fig. 7. Threshold, profit and acceptance percentage ($\gamma = 0.003$, $\alpha = 0.02$, $p = 3$ and $r_0(x) = 80 - 2.5x - 0.2x^2$).

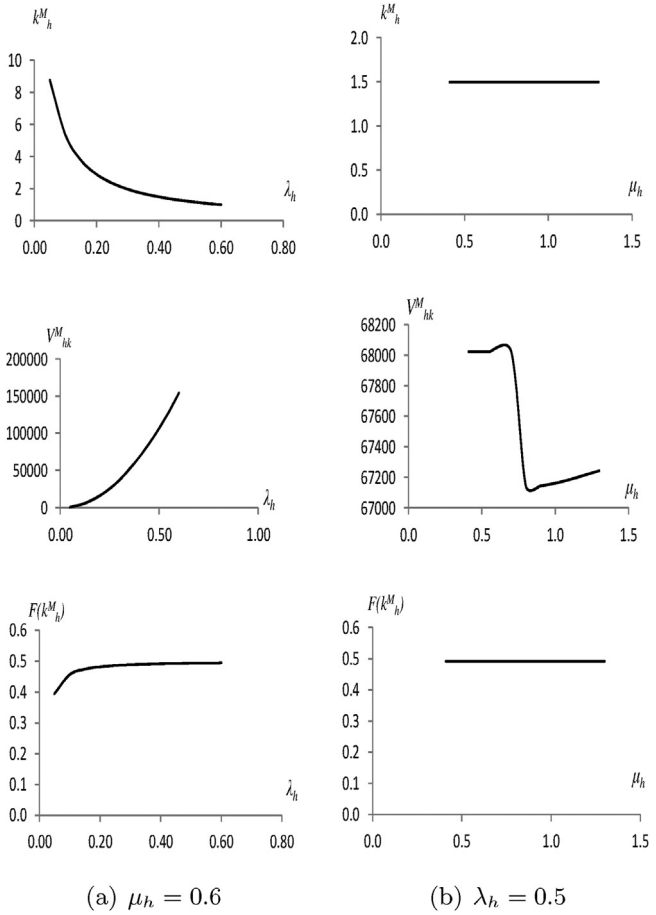


Fig. 8. Threshold, profit and acceptance percent for determined value of $\gamma = 0.003$, $\alpha = 0.02$, $p = 3$ and $r_0(x) = 80 - 2.5x - 0.2x^2$.

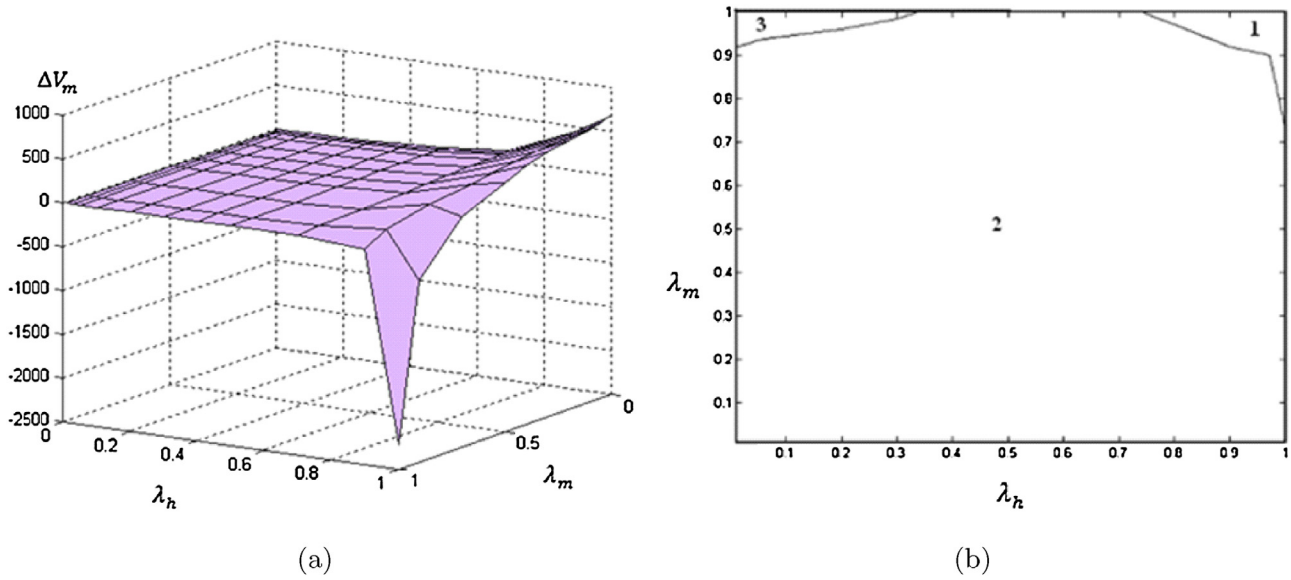


Fig. 9. Profit variability for returns from Market M, in the merged capacity configuration.

Table 2
Impact of cost-time curves ($\mu_m = 0.6$, $\lambda_m = 0.4$, $p = 3$).

	(b_1, b_2)			
	$(-5, -1.2)$	$(-4, -0.8)$	$(-3, -0.5)$	$(-2.5, -0.2)$
k	4.05	4.78	5.88	8.02
$F(k)$	0.941	0.965	0.984	1.00
$V(k)$	1349.40	1391.00	1432.60	1465.80

The cost–time curves impacts on the profit and the threshold value are shown in Table 2. The variability cost–time curves are modeled by two parameters (b_1, b_2). The general form of this curve is$r_0(x) = b_2x^2 + b_1x + b_0$. Recall that Fig. 5 shows the different kinds of cost-time curves. The profit increases with a decreasing in the value of cost-time curves. The amount of the admission value (k_i) is enhanced by an increasing in the parameter of the cost-time curves. In the last values of the parameter in the curve, 100% of the returned products are accepted under the threshold 8.02 and then the profit is maximized among other values. It is important to mention that the shape of the cost-time curves is an exogenous factor which cannot be controlled by the system manager.

The same analysis is done for returned products from Market H. The sensitivity analysis of the threshold value and the expected profit function (Eq. (9)) versus the system parameters is shown in Fig. 8a and b with $\mu_h = 0.6$ and $\lambda_h = 0.5$. We observe that an increase in the returned products rate with a constant remanufacturing rate leads to a decrease in the threshold value (Fig. 8a-1), an increase in the expected profit and the acceptance percentage (Fig. 8b-2 and a-3). Moreover, Fig. 8b-1 and b-3 reveal that the threshold values and the acceptance percentage for different values of the remanufacturing rate are approximately constant. For some critical value of the remanufacturing rate, the expected profit is collapsed (Fig. 8b-2), while the trend is ascending. This critical value has an important role in a situation of an increasing dedicated remanufacturing capacity.

The results of the sensitivity analysis for α and γ are shown in Table 3. The threshold, the acceptance percentage and the expected total discounted profit for each value of γ and α are shown as $(k^*, F(k^*), V(k^*))$. For constant decay rate and increasing discount rate, the expected discounted profit decreases. The result shows that the low discount and decay rates have more profit for the remanufacturing system. The threshold value and the percentage of the accepted returned products do not have a sensitivity in γ and α.

The cost-time curves impacts on the profit and the threshold are shown in Table 4. The variability of cost-time curves are modeled

Table 3
Impact of discount and decay rates ($\lambda_h = 0.6$ and $\mu_h = 0.8$).

	γ			
α	0.003	0.005	0.007	0.009
0.01	(1.01,0.495,551190)	(1.01,0.495,287050)	(1.01,0.495,181190)	(1.01,0.495,126280)
0.02	(1.01,0.495,313900)	(1.01,0.495,173530)	(1.01,0.495,114940)	(1.01,0.495,83360)
0.03	(1.01,0.495,220430)	(1.01,0.495,12480)	(1.01,0.495,84512)	(1.01,0.495,62455)
0.05	(1.01,0.495,139330)	(1.01,0.495,80681)	(1.01,0.495,55691)	(1.01,0.495,41910)

Table 4
Impact of cost-time curves ($\mu_h = 0.6$, $\lambda_h = 0.4$, $p = 3$).

	(b_1, b_2)			
	$(-5, -1.2)$	$(-4, -0.8)$	$(-3, -0.5)$	$(-2.5, -0.2)$
k	0.9842	0.9925	1.0002	1.0051
$F(k)$	0.4881	0.4908	0.4934	0.495
$V(k)$	308,500	310,650	312,700	313,900

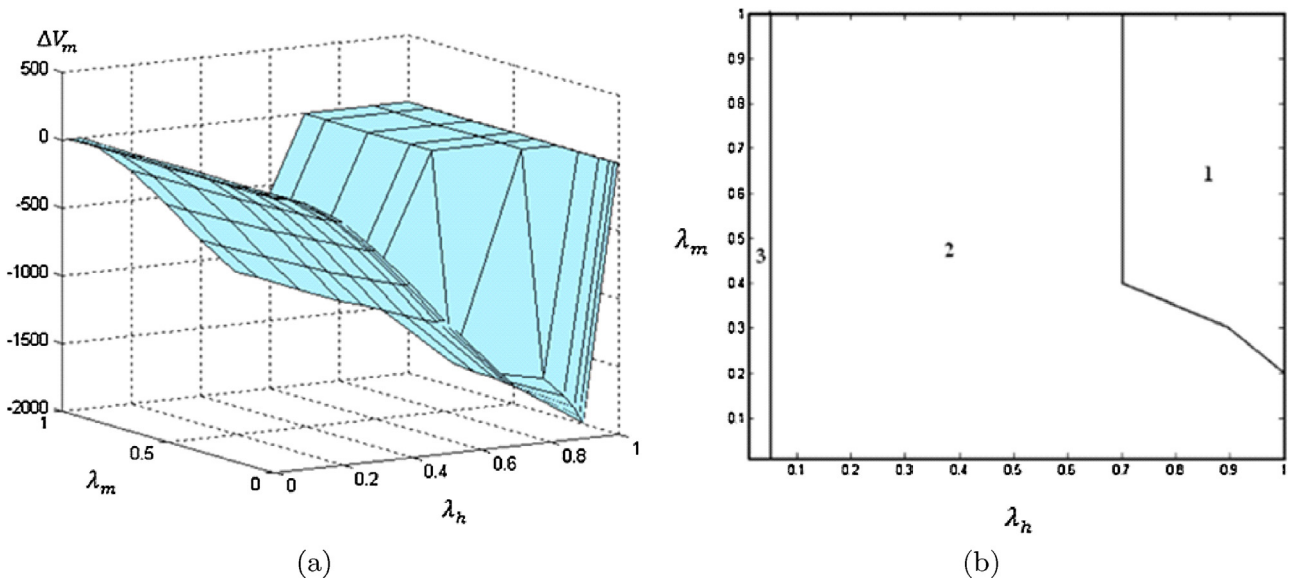


Fig. 10. Profit variability for returns from Market H, in the merged capacity system configuration.

by two parameters(b_1, b_2). A general form of this curve is $r_0(x) = b_2x^2 + b_1x + b_0$. Fig. 5 shows the different kinds of cost-time curves.

5.2. Analysis of the merged system

In each production system, the gained profit from processes is a critical criterion for the decision making. This section is devoted to analyze whether merging remanufacturing capacity can increase the profit at each returned product stream and to determine the Pareto improving region. For this purpose a key performance measure is defined as the expected profit improvement, $\Delta V = V_{merged} - V_{seperated}$.

We examine the sensitivity analysis of Market M by the performance indicator ΔV as it moves from a separated system to a merged system. By moving to a merged system, Market M sees additional remanufacturing capacity but has to share that capacity with some load from Market H. In Fig. 9, the profit variability for returned products from Market M when considering different values of λ_m and λ_h in the merged system is shown. The results show that irrespective of relative loads, Market M often sees an improvement in its performance measures.

In Fig. 9, Region 1 has $\Delta V < 0$ and Market M prefers a separated capacity. For a large value of returned product rate from Markets M and H (approximately λ_m, $\lambda_h > 0.8$), it is preferred for Market M to merge the capacity of the remanufacturing facility. So, when the capacity of remanufacturing is merged and the variability of the two markets is high, Market M prefers to remanufacture its returned products under a separated capacity system. The performance measure of Region 2 is positive ($0 < \Delta V < 500$). Region 3 has high positive performance measures ($\Delta V > 500$).

Fig. 10 shows the sensitivity analysis of Market H through the performance measure ΔV as it moves from a separated system to a merged system. For a small value of the returned products rate from Market H ($\lambda_h < 0.05$), Market H has no improvement in the expected profit (performance measure $\Delta V = 0$). In Fig. 10, this region is illustrated by Region 3. In this region, merged and

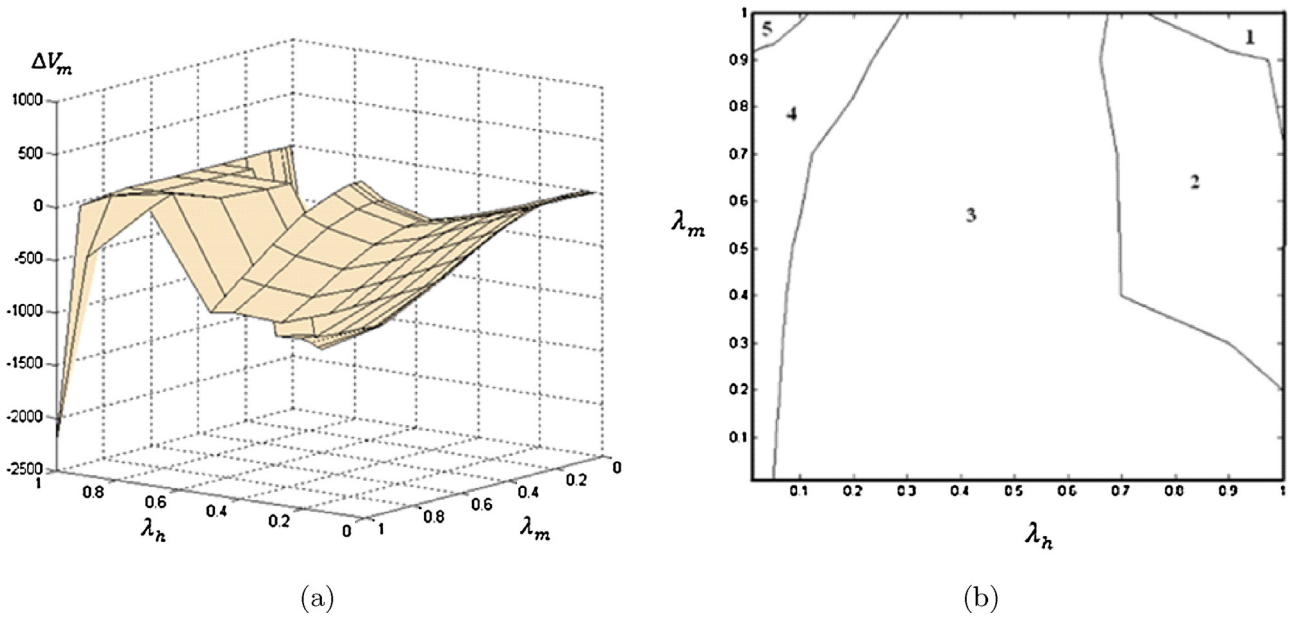


Fig. 11. The difference between gained expected profit for merged and separated capacity.

separated capacities have equal profit. Also Region 1 has no preference for merging or separating. The performance measure in Region 3 is negative ($\Delta V < 0$). Therefore, a septated capacity is preferred for Market H.

The Pareto-improving criterion requires that neither returned product streams from the markets should see a profit decrease and at least one should see a profit increase. The results shown in Figs. 9 and 10 show that merging generates Pareto-improving benefits in most of the cases. In our proposed model there is a single server remanufacturing. From the point of view of the remanufacturing facility, there is one beneficiary of the two streams; therefore the profit of the remanufacturing is the sum of the two profits. The expected profit improvement of each market can be seen as a whole ($\Delta V_m + \Delta V_h$). Fig. 11 shows the difference between the expected profit for merged and separated capacity configurations.

The performance measures of Regions 1 and 3 are negative and the remanufacturing system prefers a separated capacity. Region 2 has $\Delta V = 0$, therefore, merged and separated capacities have no impact on profit. For small values of returned product rate from Market H (approximately $\lambda_h < 0.2$), the remanufacturing system prefers to merge the capacity which is shown by Regions 4 and 5. Region 5 is the area with low values of the returned products rates from Market H ($\lambda_h < 0.1$) and high values of the returned products rate from Market M ($\lambda_m > 0.9$). In this region, the improvement in the expected profit in merged capacity is greater than 500.

6. Conclusions and future research

A remanufacturing system is considered to analyze and optimize a type of short life cycle products with stochastic serviceable products demand and stochastic processes of returned products. High congestion of returned products at the remanufacturing facility leads to a substantial delay and consequently remarketing value losses for time-sensitive products and high-tech products with short life cycles, such as electronic equipments.

The remanufacturing process was modeled by the M/M/1, H/M/1, and H_2M/M/1 queueing systems, which led us to two new lessons. First, determining the admission decision threshold value which decides about the acceptance of the returned products based on the quality and the processing time. The objective is being to maximize the total expected profit of the remanufacturing system. Second, the H_2M model shows that the difference in variability of arrivals has a significant impact on the value of merging capacity. Our analysis of the H_2M model allows us to study the interaction between the overall utilization and the arrival variability. This basic understanding of the impact of variability on merging value will be helpful for managers planning to merge the production capacities. We have also addressed the question of when does merging generate Pareto-improving benefits over the separated system. The analysis was illustrated through a numerical study. The results show the significant impact of the model parameters on the admission decision and the total expected discounted profit. Moreover, we have compared between the total expected discounted profits under situations of separated and merged capacities.

In a future research, it would be interesting to consider queueing capacity constraints for the returns. One would also include an inventory cost analysis for the warehousing of remanufactured products. Another interesting but at the same time challenging future direction is to consider a capacity constraint for the testing stage and also more general arrival processes for returns.

References

[1] Aksoy H, Gupta S. Buffer allocation plan for a remanufacturing cell? Comput Ind Eng 2005;48(3):657–77.

[2] Albin S. Approximating a point process by a renewal process, II: Superposition arrival processes to queues. Oper Res 1984;32(5):1133–62.

[3] Altiok T. Approximate analysis of queues in series with phase-type service times and blocking? Oper Res 1989;37(4):601–10.

[4] Aras N, Boyaci T, Verter V. The effect of categorizing returned products in remanufacturing? IIE Trans 2004;36(4):319–31.

[5] Bayindir Z, Erkip N, Gullu R. A model to evaluate inventory costs in a remanufacturing environment? Int J Prod Econ 2003;81(3):597–607.

[6] Bayindir Z, Erkip N, Gullu R. Assessing the benefits of remanufacturing option under one-way substitution? J Oper Res Soc 2004;56(3):286–96.

[7] Bayindir Z, Erkip N, Gullu R. Assessing the benefits of remanufacturing option under one-way substitution and capacity constraint? Comput Oper Res 2007;34(2):487–514.

[8] Behret H, Korugan A. The impact of quality uncertainty of returns in remanufacturing. In: Proceedings of the 35th international conference on computers & industrial engineering. 2005.

[9] Behret H, Korugan A. Performance analysis of a hybrid system under quality impact of returns? Comput Ind Eng 2009;56(2):507–20.

[10] Ching W. Iterative methods for queuing and manufacturing systems. Springer Monographs in Mathematics. London: Springer-Verlag; 2001.

[11] Ching W, Li T, Xue G. On hybrid re-manufacturing systems: a matrix geometric approach. In: Proceedings of the international conference on industrial engineering and systems management. 2007.

[12] Ching W, Yuen W, Loh A. An inventory model with returns and lateral transshipments? J Oper Res Soc 2003;54(6):636–41.

[13] Dekker R, Fleischmann M, Inderfurth K. Reverse logistics: quantitative models for closed-loop supply chains. Springer: Berlin; 2004.

[14] Dobos I, Richter K. A production/recycling model with quality consideration? Int J Prod Econ 2006;104(2):571–9.

[15] Eisenstein D, Iyer A. Separating logistics flows in the Chicago public school system? Oper Res 1996;44(2):265–73.

[16] Ferguson M, Guide V, Koca E, Souza G. Remanufacturing planning with different quality levels for product returns; 2006. Robert H. Smith School Research Paper No. RHS06-050.

[17] Fisher M. What is the right supply chain for your product? Harv Bus Rev 1997;75(2):105–17.

[18] Flapper S, Gayon J, Lim L. On the optimal control of manufacturing and remanufacturing activities with a single shared server? Eur J Oper Res 2014;234(1):86–98.

[19] Galbreth M, Blackburn J. Optimal acquisition and sorting policies for remanufacturing? Prod Oper Manag 2006;15(3):384–92.

[20] Geyer R, van Wassenhove LN, Atasu A. The economics of remanufacturing under limited component durability and finite product life cycles? Manag Sci 2007;53(1):88–100.

[21] Golany B, Yang J, Yu G. Economic lot-sizing with remanufacturing options? IIE Trans 2001;33(11):995–1003.

[22] Govil M, Fu M. Queueing theory in manufacturing: a survey? J Manuf Syst 1999;18(3):214–40.

[23] Gross D, Shortle J, Thompson J, Harris C. Fundamentals of queueing theory. New York: John Wiley & Sons; 2013.

[24] Guide V, Gunes E, Souza G, van Wassenhove L. The optimal disposition decision for product returns? Oper Manag Res 2008;1(1):6–14.

[25] Guide V, Souza G, van der Laan E. Performance of static priority rules for shared facilities in a remanufacturing shop with disassembly and reassembly? Eur J Oper Res 2005;164(2):341–53.

[26] Guide V, Souza G, van Wassenhove L, Blackburn J. Time value of commercial product returns? Manag Sci 2006;52(8):1200–14.

[27] Guide V, Teunter R, van Wassenhove L. Matching demand and supply to maximize profits from remanufacturing? Manuf Serv Oper Manag 2003;5(4):303–16.

[28] Guide V, van Wassenhove L. Business aspects of close-loop supply chains. Pittsburgh: Carnegie-Bosch Institute; 2003.

[29] Guide V, van Wassenhove L. The evolution of closed-loop supply chain research? Oper Res 2009;57(1):10–8.

[30] Gupta D, Gerchak Y. Quantifying operational synergies in a merger/acquisition? Manag Sci 2002;48(4):517–33.

[31] Ha A. Optimal dynamic scheduling policy for a make-to-stock production system? Oper Res 1997;45(1):42–53.

[32] Harrison J. Dynamic scheduling of a multiclass queue: discount optimality? Oper Res 1975;23(2):270–82.

[33] Inderfurth K. Impact of uncertainties on recovery behavior in a remanufacturing environment: a numerical analysis? Int J Phys Distrib Logist Manag 2005;35(5):318–36.

[34] Inderfurth K, van der Laan E. Leadtime effects and policy improvement for stochastic inventory control with remanufacturing? Int J Prod Econ 2001;71(1):381–90.

[35] Iyer A, Jain A. The logistics impact of a mixture of order-streams in a manufacturer-retailer system? Manag Sci 2003;49(7):890–906.

[36] Iyer A, Jain A. Modeling the impact of merging capacity in production-inventory systems? Manag Sci 2004;50(8):1082–94.

[37] Jain A. Value of capacity pooling in supply chains with heterogeneous customers? Eur J Oper Res 2007;177(1):239–60.

[38] Jin X, Hu S, Ni J. Assembly strategies for product remanufacturing with variable quality returns. IEEE Trans Autom Sci Eng 2012.

[39] Jin X, Ni J, Hu S, Xiao G, Biller S. Performance analysis of a remanufacturing system with warranty return admission. University of Michigan; 2012. Working paper.
[40] Karamouzian A, Teimoury E, Modarres M. A model for admission control of returned products in a remanufacturing facility using queuing theory? Int J Adv Manuf Technol 2011;54(1–4):403–12.
[41] Ketzenberg M, Souza G, Guide V. Mixed assembly and disassembly operations for remanufacturing? Prod Oper Manag 2003;12(3):320–35.
[42] Kiesmüller GP, van der Laan EA. An inventory model with dependent product demands and returns? Int J Prod Econ 2001;72(1):73–87.
[43] Koren Y. The global manufacturing revolution: product–process–business integration and reconfigurable systems. New Jersey: John Wiley & Sons; 2010.
[44] Korugan A, Gupta S. A multi-echelon inventory system with returns? Comput Ind Eng 1998;35(1):145–8.
[45] Lee H, Tang C. Modelling the costs and benefits of delayed product differentiation? Manag Sci 1997;43(1):40–53.
[46] Lund R, Mundial B. Remanufacturing: the experience of the United States and implications for developing countries. Washington, DC: World Bank; 1984.
[47] Mahadevan B, Pyke D, Fleischmann M. Periodic review, push inventory policies for remanufacturing. Eur J Oper Res 2003;151(3):536–51.
[48] Narus J, Anderson J. Rethinking distribution: adaptive channels? Harv Bus Rev 1996;74(4):112–20.
[49] Perez A, Zipkin P. Dynamic scheduling rules for a multiproduct make-to-stock queue? Oper Res 1997;45(6):919–30.
[50] Pokharel S, Mutha A. Perspectives in reverse logistics: a review. Resour Conserv Recycl 2009;53(4):175–82.
[51] Shi J, Zhang G, Sha J. Optimal production and pricing policy for a closed loop system. Resour Conserv Recycl 2011;55(6):639–47.
[52] Souza G, Ketzenberg M, Guide V. Capacitated remanufacturing with service level constraints? Prod Oper Manag 2002;11(2):231–48.
[53] Takahashi K, Morikawa K, Takeda D, Mizuno A. Inventory control for a markovian remanufacturing system with stochastic decomposition process? Int J Prod Econ 2007;108(1):416–25.
[54] Tijms H. Stochastic modelling and analysis: a computational approach. Chichester, UK: John Wiley & Sons; 1986.
[55] Toktay L, Wein L, Zenios S. Inventory management of remanufacturable products? Manag Sci 2000;46(11):1412–26.
[56] van der Laan E, Salomon M. Production planning and inventory control with remanufacturing and disposal? Eur J Oper Res 1997;102(2):264–78.
[57] van der Laan E, Salomon M, Dekker R, van Wassenhove L. Inventory control in hybrid systems with remanufacturing? Manag Sci 1999;45(5):733–47.
[58] van der Laan E, Teunter R. Simple heuristics for push and pull remanufacturing policies? Eur J Oper Res 2006;175(2):1084–102.
[59] Whitt W. Approximations for the GI/G/m queue? Prod Oper Manag 1993;2(2):114–61.
[60] Yamada T, Mizuhara N, Yamamoto H, Matsui M. A performance evaluation of disassembly systems with reverse blocking? Comput Ind Eng 2009;56(3):1113–25.
[61] Zipkin P. Foundations of inventory management. New York: McGraw-Hill; 2000.