Int J Adv Manuf Technol (2011) 54:373–392
DOI 10.1007/s00170-010-2914-x

ORIGINAL ARTICLE

Production planning and performance optimization of reconfigurable manufacturing systems using genetic algorithm

Morteza Abbasi · Mahmoud Houshmand

Received: 31 December 2009 / Accepted: 23 August 2010 / Published online: 14 September 2010

Abstract To stay competitive in the new dynamic market having large fluctuations in product demand, manufacturing companies must use systems that not only produce their goods with high productivity but also allow for rapid response to market changes. Reconfigurable manufacturing system (RMS) is a new paradigm that enables manufacturing systems to respond quickly and cost effectively to market demand. In other words, RMS is a system designed from the outset, for rapid changes in both hardware and software components, in order to quickly adjust its production capacity to fluctuations in market demand and adapt its functionality to new products. The effectiveness of an RMS depends on implementing its key characteristics and capabilities in the design as well as utilization stage. This paper focuses on the utilization stage of an RMS and introduces a methodology to effectively adjust scalable production capacities and the system functionalities to market demands. It is supposed that arrival orders of product families follow the Poisson distribution. The orders are lost if they are not met immediately. Considering these assumptions, a mixed integer nonlinear programming model is developed to determine optimum sequence of production tasks, corresponding configurations, and batch sizes. A genetic algorithm-based procedure is used to solve the model. The model is also applied to make decision on how to improve the performance of an RMS. Since there is no practical RMS, a numerical example is used to validate the results of the proposed model and its solution procedure.

M. Abbasi · M. Houshmand (✉)
Department of Industrial Engineering,
Sharif University of Technology,
Azadi Road, P. O. Box 11155-9414, Tehran, Iran
e-mail: hoshmand@sharif.edu

M. Abbasi
e-mail: abbasi@alum.sharif.edu

Keywords Reconfigurable manufacturing system · Production planning · Stochastic demand · Genetic algorithm

1 Introduction

Nowadays, manufacturing entered a new era in which all manufacturing enterprises must compete in a global economy. In the global competition, existence of numerous competitors and frequent introduction of new products cause large fluctuations in product demand. As a result, production of right products with low cost and high quality is not sufficient to success, and the new requirements such as production responsiveness and flexibility should be considered to respond rapidly to market changes and consumer needs. Accordingly, a new manufacturing capability that allows for a quick production launch of new products, with production quantities that might unexpectedly vary, became a necessity. Reconfigurable manufacturing system (RMS) is a new paradigm that offers this capability [1].

RMS is a system, designed from the outset, for rapid changes in both hardware and software components, in order to quickly adjust its production capacity to fluctuations in market demand and adapt its functionality to new products [1]. For a manufacturing system to be readily reconfigurable, the system must possess certain key characteristics. These include modularity of all system components, integrability for both ready integration and future introduction of new technology, convertibility among existing products and future products, diagnosability of the source of quality and reliability problems, and customization of hardware and controls to match the product family [2]. A system that exhibits these characteristics will allow dramatic reduction in launch time of both new systems and rebuilt systems, and achieve system upgrading relatively quickly and inexpensively by upgrading one or more modules at a time rather than replacing the entire system [3].

Reconfigurable manufacturing systems are designed and operated according to a set of basic principles which are given below [1].

1. The RMS contains adjustable production resources to respond to imminent market needs.
   - The RMS capacity is rapidly scalable in small, optimal increments.
   - The RMS functionality is rapidly adaptable to the production of new products.
2. The RMS is designed around a part/product family, with just enough customized flexibility needed to produce all members of that family.
3. To enhance the responsiveness of a manufacturing system, RMS key characteristics should be embedded in the whole system as well as in its components (mechanical, communications, and controls).
4. The RMS contains an economical mix of flexible and reconfigurable equipment with customized flexibility, such as reconfigurable machines whose functionality and productivity can be readily changed when needed.
5. In general, systems with a large number of alternative routes to produce a part are more reconfigurable, but they require higher investment cost in tooling and in material-handling systems.
6. The RMS possesses hardware and software capabilities to respond cost effectively to unpredictable events (market changes and machine failure).
7. The RMS possesses cost-effective safety capacity and stand-by functionality that is utilized to cope with unpredictable events.

The first three principles are the core principles that define a reconfigurable system. The others are secondary principles that assist in designing a cost-effective RMS.

The effectiveness of an RMS depends on implementing these principles in the design as well as utilization stage. This paper focuses on the utilization stage of an RMS and introduces a mathematical model to manage and evaluate effectiveness of RMS. This model considers the key characteristics and capabilities of RMS to adjust scalable production capacities and the functionality of the system to respond rapidly to market demands and fulfill productivity.

2 Literature review

In RMS, the required products are classified into several product families, each of which is a set of similar products [4–7]. Corresponding to each product family, there are several feasible configurations [8, 9].These feasible configurations possess different production speeds, production costs, and changeover costs. Any time the manufacturer selects a product family as a production task, RMS produces a number of products belonging to the selected family in a selected configuration. On selling a product, a reward is earned. In the completion of a production task, the manufacturer must select a family as the subsequent production task, and so on. A changeover cost is incurred when the configuration changes from one to another [1, 4, 10].

Previously, a few researches on modeling of RMS have been published [4, 11]. Zhao et al. [4] have developed the first stochastic model of an RMS that gave a thorough insight in modeling RMS. Their particular research concerns with the following three important factors in a successful RMS implementation: the optimal configurations in the design stage, optimal selection policy in the utilization stage, and increasing the performance measures of the system in the improvement. In their first paper, a stochastic model of RMS is proposed and two case studies are evaluated. Their second paper proposed an algorithm to choose the optimal configuration for production of a product family in order to maximize the average profit in the infinite horizon [5]. Their third paper focused on the optimal selection policy [6]. The last one focused on the system's performance measures, allowing the manufacturer to optimize the maximum number of orders that can be accepted where there is a high fluctuation in the market demand [7].

To avoid a very complex stochastic model and time-consuming analysis, Zhao et al. encountered the following shortcomings. According to Zhao's model [5], in the design stage, only one of the feasible configurations of a product family is selected and fixed as optimum configuration in utilization stage. That means the full capabilities of RMS such as alternative routes, scalability, and flexibility have not been considered properly. Takahashi et al. directly extended Zhao's model to remove this limitation [11]. Moreover, RMS always reacts to arrival orders and all of them face with unpredictable delay. In other words, RMS always faces with some backorders while no additional costs are charged. On the other hand, if there is no arrived order, RMS stops and waits for an arrival order that next produces as many as ordered disregarding changeover time and cost. These assumptions are not suitable in some competitive markets in which responsiveness is the main key issues, and some arrival orders may be missed, when they are not fulfilled immediately. In addition, they ignored the required time for changing the configuration of system from one to another [4].

Abbasi and Houshmand [12] have developed the first mathematical model where arrival orders follow the Poisson distribution and the orders are lost, if they cannot be met or fulfilled immediately. They used some properties of the Poisson distribution to estimate the effects of stochastic orders on the important variables of RMS production planning. Using these estimations, a mixed integer nonlinear model was developed. To solve the proposed model, a Tabu search-based procedure was applied to determine the optimum or near to optimum sequence of production tasks, corresponding product families, configurations, and batch sizes. In each iteration of this procedure, the number of production tasks contributing in the sequence was increased step by step. Then, a Tabu search algorithm was applied to propose different sequence of production tasks. Considering the proposed sequence, the model would be simplified, and it could be solved by LINGO. The goal of simplified model was to determine the optimum batch sizes and the fitness function of the proposed sequence [12].

The objective function of Abbasi and Houshmand's model [12] is total selling prices minus total production costs. This value will be unbounded, if the planning horizon is not restricted. Accordingly, they set an upper bound on the total time required to complete a solution such as a working day or week. To judge precisely among different solutions, the fitness functions of them are divided by their time length. But, to avoid complexity of analyzing a fractional objective function, this judgment criterion is not applied in the process of determining the optimum batch sizes and the fitness functions. All of these assumptions may have a significant effect on the quality of results.

The paper focuses on Abbasi and Houshmand's research [12]. In this paper, the proposed mathematical model and its solution procedure are improved. Therefore, the previous objective function is replaced by the rate of earned profit. Consequently, the applied constraint on the planning horizon can be removed. Accordingly, a precise criterion is applied in the process of determining the optimum batch sizes. In this research, a genetic-based algorithm is applied in the optimization procedure. To solve the new model having a fractional objective function, two algorithms are developed. Finally, this model and its solution procedure are used to evaluate and make decisions on how to improve the RMS performances.

In the following sections, an RMS modeling procedure is introduced. In this modeling procedure, firstly the effects of stochastic arrival orders on on-hand inventory levels, inventory holding costs, and sales are evaluated and estimated. Then, using the estimating equations, a mathematical model is developed. To solve the model, a genetic algorithm-based procedure is introduced. Since there is no actual RMS, a numerical example is used to evaluate and examine this modeling approach and its solution procedures. Finally, this methodology is applied to evaluate the important performance parameters of a sample RMS.

3 Modeling of RMS

3.1 Problem description

Consider an RMS assigned to manufacture a collection of product families. There is a set of feasible configurations corresponding to each product family having different production rates and costs. Where a product family is selected as a production task, the RMS produces a number of them in a feasible configuration. In the completion of a production task, the manufacturer must select a family as the subsequent production task, and so on. Changeover cost and time are incurred when the configuration changes from one to another. Arrival rate of orders belonging to each product family follows the Poisson distribution, and the orders are missed where they cannot be met immediately. The following definitions are used throughout this paper.

Run	Completing a production task in a specified configuration and reconfiguring it to the next production task.
Arrangement	An arrangement consists of a number of successive runs; when the final production task is completed, the system configuration should be changed to the first production task's configuration. An arrangement is defined by the number of its runs, the sequence of selected product families, the selected configurations, and the batch sizes.
Length of an arrangement	A period of time during which the arrangement is completed.

Figure 1 illustrates an arrangement with R runs. In the first runs, v_1 units of kth product family using its lth feasible configuration are produced. In the Rth production task, v_R units of mth product family are produced using its nth feasible configuration. In the final run, the system is reconfigured to its first configuration.

The fundamental objective of this paper is to determine the optimum arrangement that maximizes total profit subject to the system conditions. Total profit is equal to earned selling prices minus production costs. Production costs consist of operating costs, changeover costs, and inventory holding costs. The optimum arrangement concerns with optimum number of runs, optimum sequence of production tasks, and optimum batch sizes. The optimum arrangement adjusts production outputs to arrival orders in such a way to cover maximum orders in minimum costs.

3.2 Proposed modeling approach

Considering the conditions discussed in Section 3.1, development of a stochastic model to determine the optimum arrangement and to analyze the system's behavior is too

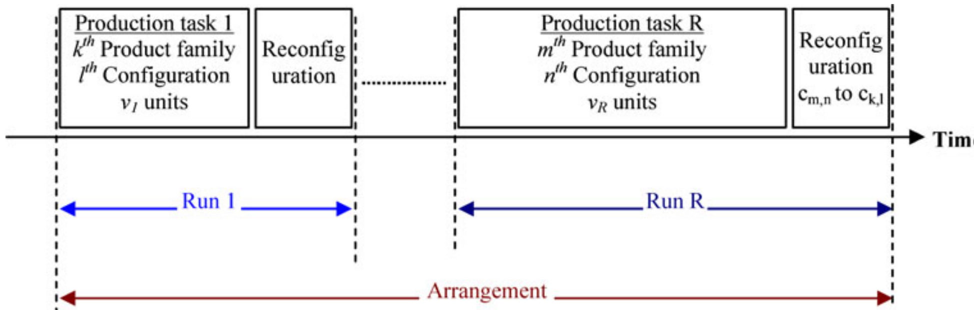


Fig. 1 Schematic presentation of production task, run, and arrangement

complex and difficult. Therefore, in this paper, to achieve a good solution through reasonable time, a mathematical model is developed.

It is not possible to use stochastic parameters directly in mathematical models. In this paper, the on-hand inventory levels, inventory holding costs, and sales are evaluated and estimated according to stochastic orders. These estimated values are used in a mixed integer nonlinear program (MINLP) to determine optimum or near to optimum arrangements.

3.2.1 Estimating the effects of stochastic arrival orders on system's conditions

Suppose that products are classified into several product families, each of which is a set of similar products ($M=\{1, 2,..., m\}$). Considering the stochastic orders, the following theorems are presented to evaluate the behaviors of on-hand inventory levels, inventory holding costs, and sales [12].

Theorem 1 If arrival rate of orders belonging to product family i ($\in M$) follows a Poisson distribution with a rate of λ_i and arrival orders are missed, if they are not fulfilled immediately, and at each run only one product family is selected as current production task, then expected on-hand inventory level of product family i ($\in M$) at time t denoted by EI_i^t is obtained by the following equations:

1. If product family i ($\in M$) is not selected as rth production task:

$$\mathrm{EI}_i^t = \sum_{x_i^t=0}^{I_i^r} (I_i^r - x_i^t) \frac{(\lambda_i \times (t-T_\mathrm{S}^r))^{x_i^t} e^{-\lambda_i \times (t-T_\mathrm{S}^r)}}{x_i^t!} \quad \text{for } T_\mathrm{S}^r \le t \le T_\mathrm{F}^r \tag{1}$$

2. If product family i ($\in M$) is selected as rth production task:

$$\mathrm{EI}_i^t = \begin{cases} \displaystyle\sum_{x_i^t=0}^{I_i^r + D_i^r \times (t-T_\mathrm{S}^r)} (I_i^r + D_i^r \times (t - T_\mathrm{S}^r) - x_i^t) \frac{(\lambda_i \times (t-T_\mathrm{S}^r))^{x_i^t} e^{-\lambda_i \times (t-T_\mathrm{S}^r)}}{x_i^t!} & \text{for} \quad T_\mathrm{S}^r \le t \le T_\mathrm{S}^r + Q_i^r / D_i^r \\ \displaystyle\sum_{x_i^t=0}^{I_i^r + Q_i^r} (I_i^r + Q_i^r - x_i^t) \frac{(\lambda_i \times (t-T_\mathrm{S}^r))^{x_i^t} e^{-\lambda_i \times (t-T_\mathrm{S}^r)}}{x_i^t!} & \text{for} \quad T_\mathrm{S}^r + Q_i^r / D_i^r < t \le T_\mathrm{F}^r \end{cases} \tag{2}$$

where rth run starts at T_S^r and finishes at T_F^r; I_i^r is on-hand inventory level of product family i at the start of rth run; x_i^t is accumulated arrived orders from the start of rth run to t ($T_\mathrm{S}^r \le t \le T_\mathrm{F}^r$); D_i^r is production rate of product family i in rth run, and Q_i^r is the number of products belonging to family i which are produced in rth run.

Proof The arrival orders of product family i follows the Poisson distribution with a rate of λ_i; therefore, the x_i^t follows the Poisson distribution with a rate of $\lambda_i \times (t - T_\mathrm{S}^r)$ [13]. Considering some properties of the Poisson distribution and possibility of missing orders, Eqs. 1 and 2 are concluded [12]. □

Theorem 2 Considering the mentioned assumptions in Theorem 1, expected inventory holding cost of product family i at time t denoted by EIH_i^t where $T_\mathrm{S}^r \le t \le T_\mathrm{F}^r$ are obtained by the following equations.

1. If product family i ($\in M$) is not selected as rth production task:

$$\mathrm{EIH}_i^t = h_i \times \int_{T_\mathrm{S}^r}^{t} \sum_{x_i^t=0}^{I_i^r} (I_i^r - x_i^t) \frac{(\lambda_i \times (t-T_\mathrm{S}^r))^{x_i^t} e^{-\lambda_i \times (t-T_\mathrm{S}^r)}}{x_i^t!} \mathrm{d}t \quad \text{for} \quad T_\mathrm{S}^r \le t \le T_\mathrm{F}^r \tag{3}$$

2. If product family i ($\in M$) is selected as rth production task:

$$\mathrm{EIH}_i^t = \begin{cases} h_i \times \int_{T_S^r}^{t} \sum_{x_i^t=0}^{I_i^r+D_i^r\times(t-T_S^r)} \left(I_i^r + D_i^r \times (t - T_S^r) - x_i^t\right) \frac{(\lambda_i\times(t-T_S^r))^{x_i^t} e^{-\lambda_i\times(t-T_S^r)}}{x_i^t!} \mathrm{d}t & \text{for} \quad T_S^r \le t \le T_S^r + Q_i^r/D_i^r \\ h_i \times \int_{T_S^r}^{T_S^r+Q_i^r/D_i^r} \sum_{x_i^t=0}^{I_i^r+D_i^r\times(t-T_S^r)} \left(I_i^r + D_i^r \times (t - T_S^r) - x_i^t\right) \frac{(\lambda_i\times(t-T_S^r))^{x_i^t} e^{-\lambda_i{}^*(t-T_S^r)}}{x_i^t!} \mathrm{d}t + & \text{for} \quad T_S^r + Q_i^r/D_i^r < t \le T_F^r \\ h_i \times \int_{T_S^r+Q_i^r/D_i^r}^{t} \sum_{x_i^t=0}^{I_i^r+Q_i^r} \left(I_i^r + Q_i^r - x_i^t\right) \frac{(\lambda_i\times(t-T_S^r))^{x_i^t} e^{-\lambda_i\times(t-T_S^r)}}{x_i^t!} \mathrm{d}t & \end{cases} \quad (4)$$

where h_i is the coefficient of inventory holding cost for product family i.

Proof Multiplying EI_i^t by h_i and dt, the expected inventory holding cost during $t\pm\varepsilon$ is obtained, where $T_S^r \le t \le T_F^r$, the integration of the $h_i \times \mathrm{EI}_i^t \times \mathrm{d}t$ can determine the expected inventory holding cost during rth run [12]. □

Theorem 3 Considering the mentioned assumptions in Theorem 1, if rth run has a length of T^r ($T^r = T_F^r - T_S^r$), then expected sold product belonging to product family i during run r denoted by ES_i^r can be obtained by following equations.

1. If product family i ($\in M$) is not selected as rth production task:

$$\mathrm{ES}_i^r = \sum_{x_i^{T^r}=0}^{I_i^r} x_i^{T^r} \frac{(\lambda_i \times T^r)^{x_i^{T^r}} e^{-\lambda_i{}^*T^r}}{x_i^{T^r}!} + \sum_{x_i^{T^r}=I_i^r+1}^{\infty} I_i^r \frac{(\lambda_i \times T^r)^{x_i^{T^r}} e^{-\lambda_i\times T^r}}{x_i^{T^r}!} \quad (5)$$

2. If product family i ($\in M$) is selected as rth production task:

$$\mathrm{ES}_i^r = \sum_{x_i^{T^r}=0}^{I_i^r+Q_i^r} x_i^{T^r} \frac{(\lambda_i \times T^r)^{x_i^{T^r}} e^{-\lambda_i\times T^r}}{x_i^{T^r}!} + \sum_{x_i^{T^r}=I_i^r+Q_i^r+1}^{\infty} (I_i^r + Q_i^r) \frac{(\lambda_i \times T^r)^{x_i^{T^r}} e^{-\lambda_i\times T^r}}{x_i^{T^r}!} \quad (6)$$

Proof Suppose that product family i is not selected as rth production task. If $x_i^{T^r} \le I_i^r$, then ES_i^r is equal to $x_i^{T^r}$. Otherwise, it is equal to I_i^r. Similarly, when product family i is selected as rth production task, if $x_i^{T^r} \le I_i^r + Q_i^r$, then ES_i^r is equal to $x_i^{T^r}$. Otherwise, it is equal to $I_i^r + Q_i^r$. Multiplying these values by corresponding probabilities and summing the results, Eqs. 8 and 9 are concluded [12]. □

Where product family i is not selected as rth production task, the continuous curves of Fig. 2a shows graphical illustration of the expected on-hand inventory level (EI_i^t; Eq. 1), and also the continuous curves of Fig. 2b shows the expected inventory holding cost (EIH_i^t; Eq. 3). All calculations are derived by Mathematica 5.1, where $T_S^r = 0$ and $T_i^r = 0$.

If product family i is not selected as rth production task, then on-hand inventory level decreases gradually and approaches zero (Fig. 2a). The slopes of graphs directly depend on the value of λ_i/I_i^r. The expected inventory holding cost increases, and eventually, it will be approximately fixed where on-hand inventory level approaches zero (Fig. 2b). If the value of λ_i/I_i^r increases, the expected inventory holding cost will decrease during rth run.

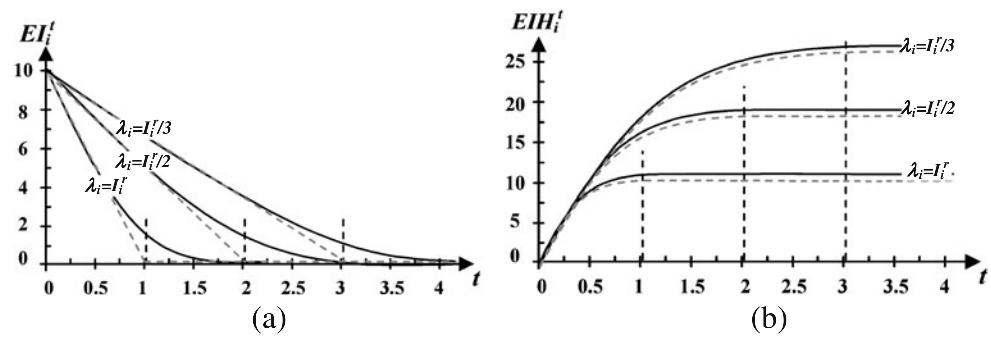
Figure 3a shows the graphical illustration of expected on-hand inventory level (EI_i^t; Eq. 2), and Fig. 3b shows expected inventory holding cost (EIH_i^t; Eq. 4) where product family i is selected as rth production task. Related calculations are derived by Mathematica 5.1, where $T_S^r = 0$ and $I_i^r = 10$.

If product family i is selected as rth production task, then the on-hand inventory level (EI_i^t) increases gradually. The slopes of curves directly depend on the value of D_i^r/λ_i. When the production task is completed, reconfiguration will be performed. During changeover time, on-hand inventory level decreases (Fig. 3a).

The above-mentioned equations are very complex, and it is not possible to use them directly in a mathematical model. Numerical experiences show that they have a simple behavior and they may be estimated with a good precision. To estimate and simplify these equations, the concept of mathematical expected value of a function is considered [13]. Estimating equations are introduced as follows.

If product family i is not selected as rth production task, the expected on-hand inventory level (EI_i^t; Eq. 1) and the expected inventory holding cost (EIH_i^t; Eq. 3) may be

Fig. 2 Graphic illustration of EI_i^t and EIH_i^t and corresponding estimating equations, if product family i is not selected as rth production task



estimated by EEI_i^t and EEIH_i^t, respectively. The estimating functions comprise of two fitting functions, and each fits a fraction of actual values. MATHEMATICA 5.1 drives the calculations and compares them [12].

$$\mathrm{EEI}_i^t = \begin{cases} I_i^r - \lambda_i \times (t - T_\mathrm{S}^r) & \text{for} \quad T_\mathrm{S}^r \le t \le T_\mathrm{S}^r + \frac{I_i^r}{\lambda_i} \\ 0 & \text{for} \quad T_\mathrm{S}^r + \frac{I_i^r}{\lambda_i} < t \le T_\mathrm{F}^r \end{cases} \tag{7}$$

$$\mathrm{EEIH}_i^t = \begin{cases} h_i \times \left(I_i^r (t - T_\mathrm{S}^r) - \frac{\lambda_i}{2} \times (t - T_\mathrm{S}^r)^2\right) & \text{for} \quad T_\mathrm{S}^r \le t \le T_\mathrm{S}^r + \frac{I_i^r}{\lambda_i} \\ h_i \times \frac{(I_i^r)^2}{2\lambda_i} & \text{for} \quad T_\mathrm{S}^r + \frac{I_i^r}{\lambda_i} < t \le T_\mathrm{F}^r \end{cases} \tag{8}$$

Figure 2a shows the actual values of expected on-hand inventory levels (EI_i^t; Eq. 1) in continuous curves and also shows the estimated values of them (EIH_i^t; Eq. 7) in dashed lines.

Maximum deviation occurs at $t = I_i^r/\lambda_i$. Numerical experiences showed that where I_i^r/λ_i increases, the maximum deviation slowly increases, but the maximum deviation divided by I_i^r decreases. For example, if $I_i^r = 10$ and λ_i=5, the maximum deviation occurs at t=2 and is equal to 1.25; and if $I_i^r = 100$ and λ_i=10, the maximum deviation occurs at t=10 and is equal to 3.99. The results show the accuracy of the estimating equations.

The continuous curves of Fig. 2b shows actual values of the expected inventory holding cost (EIH_i^t; Eq. 3), and the dashed curves show estimations of them (EEIH_i^t; Eq. 8). Maximum deviation occurs where t approaches ∞. Numerical experiences showed that where I_i^r/λ_i increases, the maximum deviation slowly increases, but the maximum deviation divided by I_i^r decreases. For example, if $I_i^r = 10$ and λ_i=5, the maximum deviation occurs where t approaches ∞ and is equal to h_i. If $I_i^r = 100$ and λ_i=10, the maximum deviation is equal to $5h_i$. The deviations are less than 1% of the actual inventory holding cost (EIH_i^t). The results show the accuracy of the estimating equations.

If product family i is selected as rth production task, EI_i^t and EIH_i^t can be estimated by the following equations [12]:

$$\mathrm{EEI}_i^t = \begin{cases} I_i^r + (D_i^r - \lambda_i) \times (t - T_\mathrm{S}^r) & \text{for} \quad T_\mathrm{S}^r \le t \le T_\mathrm{S}^r + \frac{Q_i^r}{D_i^r} \\ I_i^r + Q_i^r - \lambda_i \times (t - T_\mathrm{S}^r) & \text{for} \quad T_\mathrm{S}^r + \frac{Q_i^r}{D_i^r} < t \le T_\mathrm{F}^r \end{cases} \tag{9}$$

$$\mathrm{EEIH}_i^t = \begin{cases} h_i \times \left(I_i^r (t - T_\mathrm{S}^r) + \frac{D_i^r - \lambda_i}{2} (t - T_\mathrm{S}^r)^2\right) & \text{for} \quad T_\mathrm{S}^r \le t \le T_\mathrm{S}^r + \frac{Q_i^r}{D_i^r} \\ h_i \times \left((I_i^r + Q_i^r) \times (t - T_\mathrm{S}^r) - \frac{Q_i^{r2}}{2D_i^r} - \frac{\lambda_i}{2} (t - T_\mathrm{S}^r)^2\right) & \text{for} \quad T_\mathrm{S}^r + \frac{Q_i^r}{D_i^r} < t \le T_\mathrm{F}^r \end{cases} \tag{10}$$

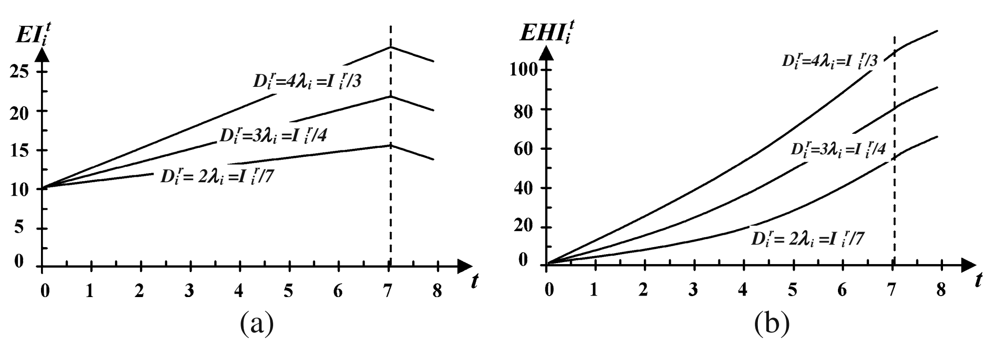


Fig. 3 Graphic illustration of EI_i^t and EIH_i^t if product family i is selected as rth production task

Actual equation of expected on-hand inventory level (EI_i^t; Eq. 2) is estimated by EEI_i^t, and actual equation of expected inventory holding cost (EIH_i^t; Eq. 4) is estimated by $EEIH_i^t$. Numerical experiences showed that deviation of estimating equations and actual equations are very small and reasonable for all of values of I_i^r, λ_i, and D_i^r.

Expected sales for product family i in rth run (ES_i^t) can be estimated by the following equations:

$$EES_i^r = \begin{cases} \lambda_i \times T^r & \text{for} \quad T^r \leq \frac{I_i^r}{\lambda_i} \quad \text{OR} \quad D_i^r > 0 \\ I_i^r & \text{for} \quad T^r > \frac{I_i^r}{\lambda_i} \end{cases} \tag{11}$$

If expected on-hand inventory level at the end of rth run ($EI_i^{T_F^r}$) is positive, then expected sales during rth run (ES_i^r) is equal to expected arrived orders ($\lambda_i \times T^r$). If $EI_i^{T_F^r}$ is equal to zero, ES_i^r is equal to I_i^r. Therefore, when the product family i is selected as rth production task, ES_i^r is equal to $\lambda_i \times T^r$.

Several numerical examples are conducted, and the behavior of equations is evaluated precisely. The results show the high efficiency of estimating equations.

It is supposed that expected on-hand inventory level at the end of each run is equal to initial on-hand inventory level at the start of the next run. Also, each run starts immediately after finishing previous one. Considering these facts, it is possible to apply the estimating equations to a set of successive runs of an arrangement. Using these equations, the model may be formulated.

Abbasi and Houshmand [12] conducted several numerical experiences to show efficiency of the proposed estimating equations. Indeed, they used the expected values of on-hand inventory level, inventory holding costs, and sales as a basis to develop a mathematical model for production planning of an RMS.

3.2.2 Modeling formulation

The model's indices are as follows:

i	Index for product families.
m	Total number of product families.
j	Index for feasible configurations for each family.
n_i	Total number of feasible configuration for each family.
$c_{i,j}$	Index for jth feasible configurations of product family $i(\in M)$ and $j=\{1,...,n_i\}$.
r	Index for production task.

The model's parameters are as follows:

S_i	Selling price of a finished product belonging to product family $i(\in M)$.
h_i	Inventory holding cost for a product belonging to product family $i(\in M)$.
λ_i	Arrival rate of orders belonging to product family $i(\in M)$.
$CR_{i,j}$	Production rate of jth feasible configuration of product family $i(\in M)$.
$CC_{i,j}$	Production cost of jth feasible configuration of product family i $(\in M)$.
$GC(c_{i,j}, c_{k,l})$	Reconfiguration cost of jth feasible configuration of product family i to be changed to lth feasible configuration of product family k, where $i,k(\in M)$, $j=\{1,...,n_i\}$ and $l=\{1,...,n_k\}$.
$GT(c_{i,j}, c_{k,l})$	Reconfiguration time of jth feasible configuration of product family i to be changed to lth feasible configuration of product family k, where $i,k(\in M)$, $j=\{1,...,n_i\}$ and $l=\{1,...,n_k\}$.
LA	Length of an arrangement.
R	Total number of production tasks to form optimum arrangement.
K_i^+	Penalty coefficient of positive gap between on-hand inventory level of product family i at the start and end of an arrangement.
K_i^-	Penalty coefficient of negative gap between on-hand inventory level of product family i at the start and end of an arrangement.
BigM	A very big value that approaches infinity.

The model's decision variables are as follows:

dev_i^+	Positive gap between on-hand inventory levels of product family i at the start and end of an arrangement.
dev_i^-	Negative gap between on-hand inventory levels of product family i at the start and end of an arrangement.
TC^r	Required time to change system configuration in rth run.

$$N_i^r = \begin{cases} 0 & \text{Where the in} - \text{hand inventory level of product family} \\ & i \text{ at the end of } r\text{th run is equal to zero} \\ 1 & \text{Otherwise} \end{cases}$$

$$M_i^r = \begin{cases} 1 & \text{Where the product family } i \text{ is selected as the} \\ & r\text{th production task} \\ 0 & \text{Otherwise} \end{cases}$$

$$P_{i,j}^r = \begin{cases} 1 & \text{Where the } j\text{th configuration of product family} \\ & i \text{ selected in the } r\text{th production task} \\ 0 & \text{Otherwise} \end{cases}$$

The model is formulated as follows.

Problem (P1):

$$\text{Maximize}\quad \Big(\sum_{r=1}^{R}\sum_{i=1}^{m} N_i^r\Big(S_i\lambda_i T^r - h_i\Big(I_i^r T^r + \Big(\frac{M_i^r D_i^r - \lambda_i}{2}\Big)T^{r2} - D_i^r \mathrm{TC}^{r2}/2\Big) + \big(1-N_i^r\big)\Big(S_i I_i^r - \frac{I_i^{r2} h_i}{2\lambda_i}\Big) - \sum_{r=1}^{R}\sum_{i}^{m}\sum_{j}^{n_i} \mathrm{CC}_{i,j} P_{i,j}^r Q_i^r - \sum_{r=1}^{R}\sum_{i=1}^{m}\sum_{j=1}^{n_i}\sum_{k=1,k\neq i}^{m}\sum_{l=1}^{n_k} P_{i,j}^r P_{k,l}^{r+1} G(c_{i,j}, c_{k,l}) - \sum_{i=1}^{m}\mathrm{dev}_i^+ K_i^+ - \sum_{i=1}^{m}\mathrm{dev}_i^- K_i^-\Big)/\mathrm{LA}$$

Subject to:

$$\sum_{i=1}^{m} M_i^r = 1 \quad \forall r = 1,\ldots,\ R \tag{12}$$

$$\sum_{j=1}^{n_i} P_{i,j}^r = M_i^r \quad \forall r = 1,\ldots,\ R;\forall i(\in M) \tag{13}$$

$$Q_i^r \le M_i^r \times \mathrm{BigM} \quad \forall r = 1,\ldots,\ R;\forall i(\in M) \tag{14}$$

$$D_i^r = \sum_{j=1}^{n_i} P_{i,j}^r \mathrm{CR}_{i,j} \quad \forall r = 1,\ldots,\ R;\forall i(\in M) \tag{15}$$

$$\mathrm{TC}^r = \sum_{i=1}^{m}\sum_{j=1}^{n_i}\sum_{k=1,k\neq i}^{m}\sum_{l=1}^{n_k} P_{i,j}^r P_{k,l}^{r+1}\mathrm{GT}(c_{i,j}, c_{k,l}) \quad \forall r = 1,\ldots,\ R \tag{16}$$

$$T^r = \frac{\sum_{i=1}^{m} Q_i^r}{\sum_{i=1}^{m} D_i^r} + \mathrm{TC}^r \quad \forall r = 1,\ldots,\ R \tag{17}$$

$$T^r + N_i^r \times \mathrm{BigM} \ge \frac{Q_i^r + I_i^r}{\lambda_i} \quad \forall r = 1,\ldots,\ R;\forall i(\in M) \tag{18}$$

$$T^r \le (1 - N_i^r) \times \mathrm{BigM} + \frac{Q_i^r + I_i^r}{\lambda_i} \quad \forall r = 1,\ldots,\ R;\forall i(\in M) \tag{19}$$

$$I_i^{r+1} = N_i^r(I_i^r + Q_i^r - \lambda_i T^r) \quad \forall r = 1,\ldots,\ R;\forall i(\in M) \tag{20}$$

$$I_i^1 - I_i^{R+1} + \mathrm{dev}_i^+ - \mathrm{dev}_i^- = 0 \quad \forall i(\in M) \tag{21}$$

$$P_{i,j}^{R+1} = P_{i,j}^1 \quad \forall i,j(\in M) \tag{22}$$

$$\mathrm{LA} = \sum_{r=1}^{R} T^r \tag{23}$$

$$Q_i^r \ge 0,\ \text{Integer} \quad \forall r = 1,\ldots,R;\forall i(\in M) \tag{24}$$

$$D_i^r \ge 0 \quad \forall r = 1,\ldots,\ R;\forall i(\in M) \tag{25}$$

$$I_i^r \ge 0,\ \text{Integer} \quad \forall r = 1,\ldots,\ R;\forall i(\in M) \tag{26}$$

$$T^r \ge 0 \quad \forall r = 1,\ldots,\ R \tag{27}$$

$$\mathrm{TC}^r \ge 0 \quad \forall r = 1,\ldots,\ R \tag{28}$$

$$\mathrm{dev}_i^+ \ge 0,\ \text{Integer} \quad \forall i(\in M) \tag{29}$$

$$\mathrm{dev}_i^- \ge 0,\ \text{Integer} \quad \forall i(\in M) \tag{30}$$

$$M_i^r \in (0,1) \quad \forall r = 1,\ldots,\ R;\forall i(\in M) \tag{31}$$

$$N_i^r \in (0,1) \quad \forall r = 1,\ldots,\ R;\forall i(\in M) \tag{32}$$

$$P_{i,j}^r \in (0,1) \quad \forall r = 1,\ldots,\ R;\forall i(\in M) \tag{33}$$

The goal of model is to maximize the total sales minus the total costs over the length of arrangement. The total costs comprised of inventory holding costs, production costs, and reconfiguration costs. Indeed, the objective function represents the rate of earned benefit for a specified arrangement. The earned profit is divided by the length of arrangement to avoid unbounded solutions. Also, this objective function is a good judgment criterion where comparing the arrangements having different length time.

The constraints of model are as follows. In each run, only one of the product families can be selected as current production task (constraint 12). If product i is selected as rth production task, only one of its feasible configurations must be selected as the system configuration (constraint 13). If product family i is not selected as rth production task, then $Q_i^r = D_i^r = 0$. If product family i is selected as rth production task, its production rate is equal to the production rate of the selected configuration and $Q_i^r \ge 0$ (constraints 14 and 15). The changeover time in each run depends on the current and subsequent configuration (constraint 16). Constraint 17 determines the required time to complete rth run. It is equal to production time plus changeover time. Production time is equal to batch size of rth production task divided by its production rate.

Constraints 18 and 19 are used to determine the value of N_i^r. This value determines whether the expected on-hand inventory level of product family i at the end of rth run is positive or zero. Using N_i^r and the estimated values, the following variables are obtained and included in objective function:

1. The expected inventory levels of each product family at the start of $r+1$st run (constraint 20).
2. The expected sales of each product family in rth run.
3. The expected inventory holding cost of each product family in rth run.

Constraints 21 and 22 ensure a repeatable arrangement. For each product family, gap between on-hand inventory level at the start and end of an arrangement must be minimized as much as possible. To minimize these gaps, some penalties are charged in the objective function. Constraint 21 determines these gaps. Constraint 22 changes the final production task configuration to the first configuration. Constraint 23 determines the length of arrangement. Constraints 24–33 ensure the solutions feasibilities.

4 Solution procedure

The goal of the model is to determine optimum arrangements having optimum number of runs, optimum sequence of production tasks, and optimum batch sizes. Also, each production task is defined by a product family and one of its feasible configurations. The following procedure is suggested to determine the optimum or near to optimum arrangement.

Algorithm 1:
Step A: *Set the number of runs (R) to 1.*
Step B: *Find the optimum or near to optimum arrangement with R runs.*
Step C: *Check the stopping criterion. If it is confirmed go to step F.*
Step D: *Set R to R+1 and go to step B.*
Step F: *Stop.*

If the number of runs is very large, RMS wastes a considerable portion of its production time due to changeovers. When the number of runs is low, RMS cannot effectively adapt itself to arrival orders. Therefore, in the early iterations as the number of runs increases, the objective function may be improved. However, following some iteration, the objective function may be not improved and even may converge to previous values. A stopping criterion may be defined based on this fact. For example, if the objective function of the model does not improve considerably for nth successive iterations, then stopping criterion will be confirmed.

If the batch sizes and the system configurations are not considered, the problem of the optimum arrangement will be converted to a traveling salesman problem (TSP). TSP is considered as a *NP-Complete* problem; therefore, a heuristic method is necessary to find optimum or near to optimum arrangements. In this paper, a genetic algorithm-based method is proposed to determine near to optimum arrangement where the number of runs (R) is predefined by Algorithm 1. The sequence of selecting product families and one of their feasible configurations are proposed by genetic algorithm (GA). Appling these values, problem P1 will be simplified and can be solved to determine the optimum batch size of each production task. The following procedure is suggested to determine the optimum or near to optimum arrangement where the number of runs is predefined by Algorithm 1.

Algorithm 2:
Step A: *Set the number of runs to R.*
Step B: *Candidate a population of solutions using genetic algorithm.*
Step C: *Simplify the model (P1) using one of candidate solutions and determine optimum batch sizes and the fitness function of the candidate solution.*
Step E: *If all member of population are considered, go to Step F.*
Step F: *Check the stopping criterion of genetic algorithm. If it is not confirmed go to step B.*
Step G: *Stop.*

4.1 Genetic algorithm

GA is one of the modern heuristic optimization algorithms, which is widely adopted by researchers in solving combinatorial problems. GA uses the concept of evolutionary computation imitating the natural selection and biological reproduction of animal species [14, 15]. It originates from Darwin's "survival of the fittest" concept, which means a good parent produce better offspring. GA has been successfully applied to classical combinatorial problems such as flexible job-shop scheduling [16, 17], flow-shop scheduling [18], and open-shop scheduling problem [19].

Prior to the application of GA, we need to design the genetic representation (or chromosome) of the candidate solutions. Herein, a chromosome represents each solution in the initial solution set of solutions (population). The size of the population depends on the size and the nature of the problem at hand. The chromosome evolves through a crossover operator and a mutation operator to produce children, improving on the current set of solutions. The chromosomes in the population are then evaluated through a fitness function, and the less fit chromosomes are replaced with better children. The processes of crossover, evaluation, and selection are repeated for a predetermined number of iterations called generations, usually up to the point where the system ceases to improve or the population has converged to a few well-performing chromosomes.

4.1.1 Encoding

The initial step is to design a suitable genetic representation (or chromosome) of the solutions. This is a key issue for a successful implementation of GA because it applies probabilistic transition rule on each chromosome to create a new population of chromosomes. In this paper, a chromosome defines a sequence of production tasks. A production task is defined by a product family and one of its feasible configurations. Therefore, the proposed structure for chromosomes has two sets of interrelated genes. Consider an arrangement with R runs; genetic representation of this arrangement has $2\times R$ genes in two segments as follows:

1. The product families segment.
2. The configurations segment.

In the product families segment, each gene represents one of product families that contribute in an arrangement. Also, in the configurations segment, each gene represents one of feasible configurations of the corresponding product family in the product families segment. The sequence of genes in the chromosome represents the sequence of production tasks. Table 1 illustrates a genetic representation (or chromosome) of an arrangement having R runs.

In this representation, f_i ($\in M$) illustrates the product family which is selected as ith production task, and c_{f_i} ($\in C_{f_i}$) illustrates one of the feasible configurations of the product family f_i ($\in M$), which is selected to produce a number of this product family within ith production task. Considering the problem definition and assumptions, any sequence of production tasks with feasible product family and feasible configuration can represent a feasible arrangement. Therefore, there is no restriction to determine the genes of a chromosome except feasibility of the product families and their configurations.

4.1.2 Initial population

Selecting a good initial population and reasonable population size can largely increase the efficiency of a GA. To generate a good initial population, a simple rule would be applied to generate initial population. This rule ensures that all of the product families contribute in definition of each chromosome of initial population, if it is possible.

4.1.3 Cloning operator

The cloning operator involves keeping the best solutions. In the proposed GA, this operator copies some of the current best chromosomes to the next population.

4.1.4 Parent selection operator

The parent selection operator is an important process that directs a GA search toward promising regions in a search space. Two parents are selected from the solutions of a particular generation by selection methods that assign reproduction opportunities to each individual parent in the population. For this experimentation, a binary tournament selection method was applied that began by forming two teams of chromosomes [20]. Each team consists of two chromosomes randomly drawn from the current population. The two best chromosomes that are taken from one of the two teams are chosen for crossover operations. As such, offspring are generated and enter into a new population.

Table 1 A genetic representation (or chromosome) of an arrangement with R runs

Run_1	Run_2	Run_3	...	Run_R	
f_1	f_2	f_3	...	f_R	The product families segment
c_{f_1}	c_{f_2}	c_{f_3}	...	c_{f_R}	The configurations segment

4.1.5 Crossover operator

The crossover operator generates new children by combining information contained in the chromosomes of the parents so that new chromosomes will have the best parts of the parents' chromosomes. Herein, we applied the one-point crossover which is randomly located in the product families segment of chromosome and its corresponding point in the configurations segment. The crossover operator randomly chooses two individual parents and generates a random value (r) between 1 to $R-1$ as the crossover point. As shown in Fig. 4, each chromosome of the parent is divided into two sections from crossover point. Then, the first section of a chromosome and the second section of other chromosome build one of the children. Another child is built by remained parts of parents' chromosomes. Note that all of the children generated by this operator are feasible because of feasibility of the parents.

4.1.6 Mutation operator

Mutation is used to produce small perturbations on chromosomes to promote diversity of the population and to prevent solutions from being trapped at a local optimum. After recombination, some children undergo mutation. Mutation operates with some small probability, usually from 0% to 10%. The rationale is to provide a small amount of randomness. The type of mutation varies depending on the encoding as well as the crossover. In the proposed GA, mutation is randomly performed by one of following ways that insure the feasibility of mutated children.

1. Changing the product family of a gene belonging to the product families segment and its corresponding configuration in the configurations segment.
2. Changing the configuration of a gene belonging to the configurations segment, considering the set of feasible configurations.

4.1.7 Fitness function

Each chromosome represents a candidate solution. Decoding a chromosome, the sequence of selecting the product families, and their corresponding configurations is determined. Using these values, problem P1 can be simplified and solved as described in Section 4.2. By solving the simplified model, the optimum batch sizes and the fitness function of the chromosome are determined.

4.1.8 Termination criterion

The termination criterion determines when GA will stop. In other words, the genetic operations are repeated until a termination condition is met. In this research, GA will stop if maximum number of generations, max_gen, has been executed or the pre-set number of generations without improvement in the last best solution, max_no_improve, reaches.

4.2 Solving the simplified problem

The purpose of this section is to determine the optimum batch size of each production task and the value of fitness function of a chromosome. Each chromosome represents the sequence of selecting the product families and their corresponding configurations. Using these values, most of integer variables and some of constraints of problem P1 are removed, and its objective function is simplified. To solve the simplified problem, the following two notes could be considered.

Note 1 It is assumed that in the completion of an arrangement, the same arrangement will be repeated again, and so forth. To avoid overproduction in a repeatable arrangement, the expected values of unsold products within an arrangement must be near to zero. In other words, optimum batch sizes ensure that the sum of produced product of a family is lower than its expected arrival orders.

Note 2 Suppose an arrangement with one production task where product family i is produced with its jth configuration. The production rate is $\mathrm{CR}_{i,j}$ and the finished producs are sold with a rate of λ_i where $\mathrm{CR}_{i,j} > \lambda_i$. In this situation according to the estimating equations, the objective function of the model will be concave and its maximum value occurs where production volume is as follows:

$$Q_i^* = \frac{S_i - \mathrm{CC}_{i,j}}{h_i(1/\lambda_i - 1/\mathrm{CR}_{i,j})} \tag{34}$$

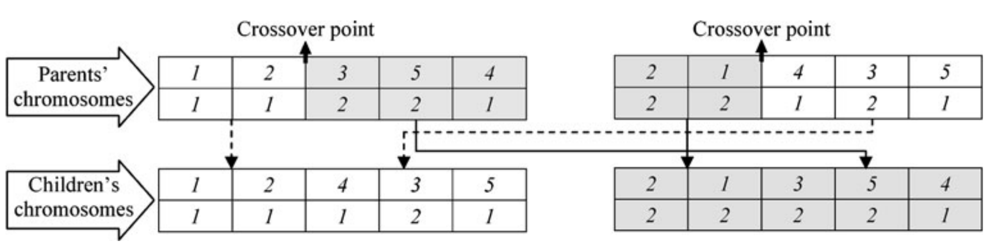


Fig. 4 Schematic illustration of one-point crossover

As shown in Fig. 5, where production volume is zero, the expected selling prices, production costs, and inventory holding costs are also equal to zero. By increasing the production volume, the expected selling prices minus production costs is increased with greater slope than the expected inventory holding costs. In the optimum production volume (Q_i^*), the expected selling prices minus production costs is two times greater than the expected inventory holding costs.

These notes represent upper bounds of the optimum batch sizes especially where several product families are to be produced by an RMS. In this research, the following two methods are developed to solve the simplified problem: The gradient method finds the optimal solution within a very short time. The time consuming numeration method is merely used to verify the optimality of the gradient method.

4.2.1 Numeration method

Suppose that q_r is the batch size of rth production task; therefore, $Q=\{q_1, q_2, ..., q_R\}$ represent a point in the state space of batch sizes of an arrangement with R runs. Also, assume that product family f_r ($\in M$) is selected as rth production task; the state space can be illustrated as follow where $Q^*_{f_r}$ is determined according to Eq. 34.

$$S = \{Q | 0 \leq q_r \leq Q^*_{f_r}, f_r \in M,\ 1 \leq r \leq R,\ q_r \text{ is integer}\} \tag{35}$$

In the numeration method, the state space is explored thoroughly, and the objective function for each member of state space is calculated, and finally the optimum solution is determined.

The state space has $\prod_{r=1}^{R} (1 + Q^*_{f_r})$ members that increased exponentially where the number of runs increases. This method may be effective where according to Eq. 34, the values of $Q^*_{f_r}$ are relatively small.

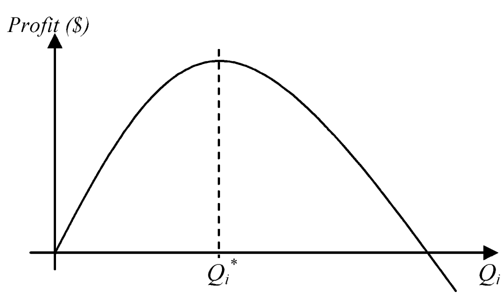


Fig. 5 Illustration of the objective function where only product family i is produced

4.2.2 Gradient method

Gradient method is based on the greedy algorithm. It starts with $Q=0=\{0, 0, ..., 0\}$. In the iterations, the value of first batch size (q_1) is increased by 1, and then the objective function is calculated. In the next step, the value of the batch size return to previous value and next batch size (q_2) is increased by 1, and the objective function is calculated again, and so on. In the iterations, one of the batch sizes having the greatest positive impact on the objective function is selected to be increased by one and fixed for the next iterations. Considering Note 1 (Section 4.2), an increasing in a batch size is confirmed where it does not affect the expected unsold products to exceed zero.

This method considers the derivation of the objective function with respect to batch sizes. As shown in Fig. 5, where the value of a batch size increases, the slope of objective function decreases, and finally where $q_r = Q^*_{f_r}$, its slope approaches zero. In the worst case, this method considers $\sum_{r=1}^{R} R(1 + Q^*_{f_r})$ states, which it is very much smaller than the value expected by the numeration method.

Numerous examples are conducted to compare the results of each method. The results show that optimum solutions of each method are the same. However, the gradient method can determine the optimum solution in a small number of iterations. These methods can be coded easily and integrated with GA effectively.

Considering that there is no available data on actual RMS, in this research, a numerical example is presented to illustrate the proposed procedures to determine the optimum or near to optimum arrangement.

5 Numerical experiences

In this section, some examples are presented to evaluate the efficiency of the proposed model and its solution procedure. First, an RMS having four product families is considered, and the optimum or near to optimum arrangement of production tasks is determined using the presented heuristic algorithms. Next, the effect of some important parameters of system such as arrival rates of orders, inventory holding cost, changeover time, and cost on the optimum arrangement will be evaluated. Now, consider an RMS having four product families. The design parameters of this RMS are shown in Table 2.

According to first step of Algorithm 1, arrangements with one run are considered. Using Eq. 34, the optimum batch size for different arrangement having one run is determined. In this iteration, only the mutation operator can be applied. The result shows that producing third product family with its first configuration is the optimum arrangement having one run. The optimum batch size is 120, and the objective function is \$10.88/min.

In the second iteration of Algorithm 1, arrangements with two runs are considered. In this iteration, the proposed GA is applied to determine the optimum or near to optimum arrangements. Where population size is 50, within a few iterations, all of the candidate solutions converge to the following chromosomes (Table 3).

Considering the problem definition and modeling assumptions such as repeatability of an arrangement, the above solutions are the same in practice. The value of their fitness functions are 34.68 and the length of corresponding arrangements is 9.46.

Where the number of runs is set to 3, the best solutions proposed by GA are shown in Table 4.

Similarly, these three alternative solutions are the same in practice. Their fitness functions and the length of corresponding arrangements are 46.33 and 16.32, respectively. In next iteration, the value of R is set to 4 and GA converges to solution shown in Table 5.

In this iteration, there are four same solutions that the fitness functions of them and the length of corresponding arrangements are equal to 54.8 and 27.20, respectively. By increasing the value of R in next iterations, the fitness function of the best solutions would not be increased and converged to the best solution having four runs. For example, where R is 5, GA converges to solutions shown in Table 6.

The arrangement shown in Table 7 is one of the proposed solutions by GA, where R is 8.

Table 2 The parameters of an RMS

Families $\{1, ..., m\}$	$\{1, 2, 3, 4\}$			
Family i	1	2	3	4
Arrival rate: λ_i	1.6	1.1	1.45	1.2
Selling price: S_i	28	30	35	32
Inventory holding cost: h_i	0. 2	0. 3	0. 2	0. 1
Feasible production configurations: $C_i=\{c_{i,1},...,c_{i,ni}\}$	$C_1=\{c_{1,1},c_{1,2}\}$	$C_2=\{c_{2,1},c_{2,2},c_{2,3}\}$	$C_3=\{c_{3,1},c_{3,2}\}$	$C_4=\{c_{4,1},c_{4,2}\}$
Production configuration cost: $CC_{i,j}$	{19, 22}	{18, 16, 20}	{20, 23}	{16, 19}
Production configuration rate: $CR_{i,j}$	{16, 12}	{15, 12, 9}	{16, 13}	{13, 10}
Changeover cost: $GC(C_{i,j},C_{k,l})$	$GC(C_{1,1}$ to $C_{k,l})=$ 0 15.7 14.8 16.5 12.7 15.1 14.0 16.7 14.3 $GC(C_{1,2}$ to $C_{k,l})=$ 12.5 14.5 13.8 16.9 0.0 14.7 13.4 17.3 15.1	$GC(C_{2,1}$ to $C_{k,l})=$ 17.1 0.0 15.1 13.5 16.4 12.2 15.7 13.1 12.4 $GC(C_{2,2}$ to $C_{k,l})=$ 17.5 12.5 15.9 12.7 16.9 0.0 15.3 13.0 12.1 $GC(C_{2,3}$ to $C_{k,l})=$ 16.0 12.4 15.4 13.8 16.4 12.0 15.2 14.5 0.0	$GC(C_{3,1}$ to $C_{k,l})=$ 15.0 18.9 0.0 13.5 15.4 18.1 12.7 15.9 18.5 $GC(C_{3,2}$ to $C_{k,l})=$ 15.7 16.6 12.5 14.9 16.1 15.9 0.0 15.4 15.2	$GC(C_{4,1}$ to $C_{k,l})=$ 13.5 16.3 14.3 0.0 14.1 16.9 14.9 12.2 17.7 $GC(C_{4,2}$ to $C_{k,l})=$ 12.8 13.1 17.7 12.1 13.0 13.7 17.2 0.0 14.4
Changeover time: $GT(C_{i,j},C_{k,l})$	$GT(C_{1,1}$ to $C_{k,l})=$ 0.0 5.7 4.8 6.5 2.7 5.1 4.0 6.7 4.3 $GT(C_{1,2}$ to $C_{k,l})=$ 2.5 4.5 3.8 6.9 0.0 4.7 3.4 7.3 5.1	$GT(C_{2,1}$ to $C_{k,l})=$ 7.1 0.0 5.1 3.5 6.4 2.2 5.7 3.1 2.4 $GT(C_{2,2}$ to $C_{k,l})=$ 7.5 2.5 5.9 2.7 6.9 0.0 5.3 3.0 2.1 $GT(C_{2,3}$ to $C_{k,l})=$ 6.0 2.4 5.4 3.8 6.4 2.0 5.2 4.5 0.0	$GT(C_{3,1}$ to $C_{k,l})=$ 5.0 8.9 0.0 3.5 5.4 8.1 2.7 5.9 8.5 $GT(C_{3,2}$ to $C_{k,l})=$ 5.7 6.6 2.5 4.9 6.1 5.9 0.0 5.4 5.2	$GT(C_{4,1}$ to $C_{k,l})=$ 3.5 6.3 4.3 0.0 4.1 6.9 4.9 2.2 7.7 $GT(C_{4,2}$ to $C_{k,l})=$ 2.8 3.1 7.7 2.1 3.0 3.7 7.2 0.0 4.4

Table 3 The proposed solutions by GA where R=2

	Production task 1	Production task 2
Product families segment	3	4
Configuration segment	1	1
Batch size	13	11
Product families segment	4	3
Configuration segment	1	1
Batch size	11	13

In practice, all of the above solutions are similar to the best solutions having four runs. Consequently, the optimum or near to optimum arrangement is described as follows:

Run 1 Producing 43 units of product family 1 using its first configuration as first production task. Then, the system's configuration is changed from first configuration of product family 1 to second configuration of product family 2.

Run 2 Producing 29 units of product family 2 using its second configuration as second production task. Then, the system's configuration is changed from second configuration of product family 2 to first configuration of product family 4.

Run 3 Producing 32 units of product family 4 using its first configuration as third production task. Then, the system's configuration is changed from first configuration of product family 4 to first configuration of product family 3.

Run 4 Producing 39 units of product family 3 using its first configuration as fourth production task. Then, the system's configuration is changed from first configuration of product family 3 to first configuration of product family 1.

In following sections, the effects of some important parameters of system on the optimum or near to optimum solutions are considered. The goal is to evaluate the proposed model and its solution procedure. Furthermore, the best alternative to improve performance of a system may be identified.

Table 4 The proposed solutions by GA where R=3

	Task 1	Task 2	Task 3
Product families segment	1	3	4
Configuration segment	1	1	1
Batch size	26	23	19
Product families segment	3	4	1
Configuration segment	1	1	1
Batch size	23	19	26
Product families segment	4	1	3
Configuration segment	1	1	2
Batch size	19	26	23

Table 5 The proposed solution by GA where R=4

	Task 1	Task 2	Task 3	Task 4
Product families segment	1	2	4	3
Configuration segment	1	2	1	1
Batch size	43	29	32	39

5.1 The effect of the orders arrival rates on the solutions

The orders arrival rates have a direct influence on the sales opportunities. Increasing these rates may cause a significant positive influence on the earned profit. However, to handle a higher order arrival rates, the capability of the system should be considered. As a result, a higher arrival orders may cause the bigger batch sizes. The bigger batch sizes can be described as the reaction of the system to increase the production capacity by reducing the rate of changeover times. If the arrival rates of orders exceed the production capacity of the system, some product families may be removed from the optimum arrangement. In this situation, the limited capacity may be dedicated to the products having a higher profit margin. Contrarily, decreasing the arrival rate of orders may cause an inverse effect on the earned profit and the batch sizes. In the mentioned RMS, following the solutions in Table 8 are determined by GA under different conditions.

Where the arrival orders were very much decreased ($\lambda_i \times 0.5$), the optimum sequence of production tasks is changed to 1, 2, 3, and 4. This change increases the total changeover times to avoid overproduction and the related excessive inventory holding costs.

Table 6 The proposed solutions by GA where R=5

	Task 1	Task 2	Task 3	Task 4	Task 5
Product families segment	1	1	2	4	3
Configuration segment	1	1	2	1	1
Batch size ($0 \le x \le 43$)	$43-x$	x	29	32	39
Product families segment	1	2	2	4	3
Configuration segment	1	2	2	1	1
Batch size ($0 \le x \le 29$)	43	$29-x$	x	32	39
Product families segment	1	2	4	4	3
Configuration segment	1	2	1	1	1
Batch size ($0 \le x \le 32$)	43	29	$32-x$	x	39
Product families segment	1	2	4	3	3
Configuration segment	1	2	1	1	1
Batch size ($0 \le x \le 39$)	43	29	32	$39-x$	x

Table 7 The proposed solution by GA where R=8

	Task 1	Task 2	Task 3	Task 4	Task 5	Task 6	Task 7	Task 8
Product families segment	1	2	4	3	1	2	4	3
Configuration segment	1	2	1	1	1	2	1	1
Batch size	43	29	32	39	43	29	32	39

Generally, if the arrival rate of orders belonging to a family is very much greater than others, this product family may be repeated in the optimum arrangement. For example, in the mentioned RMS, suppose that only arrival rate of orders belonging to product family 1 (λ_1) is duplicated, the optimum or near to optimum arrangement is as show in Table 9.

As expected, since λ_1 is approximately two times greater than the others, the best number of runs is 5 and product family 1 is selected two times in the optimum arrangement. For some greater value of λ_1, product family 1 is repeated in the optimum arrangement alternatively. Where λ_1 is decreased, the fitness function and the batch sizes decrease.

5.2 The effect of the production rates on the solutions

The production rates are considered as the capability of system to respond the market demands. The effect of production rates on the earned profit is similar to the effect of arrival rates of orders in relatively less violence. But, the production rates have an inverse effect on the batch sizes. In other words, if the production capacity of system is higher than market demand, increasing the production rates may cause to decrease the batch sizes. The smaller batch sizes are considered as a reaction of the system to decrease the production capacity and avoid overproduction and related inventory holding costs. Contrarily, if the production rates are reduced, the batch sizes are increased. The bigger batch sizes enlarge production times of an arrangement in comparison with changeover times.

In some cases, if the production rates in comparison with arrival orders decease considerably, some product families may be removed from the optimum arrangement to be dedicated limited capacity to the other more valuable products (Table 10).

Where production rates are very much higher than arrival rates of orders (D_i×5), some products may be repeated in the optimum arrangement to increase total changeover times. It is a reaction of the system to avoid overproductions and related inventory holding costs.

Table 8 The proposed solutions by GA where all of order arrival rates are multiplied by 1.5, 2, 0.8, and 0.5

Task 1	Task 2	Task 3	Task 4
Where λ_i×1.5 for i=1,…,m. The fitness function, 77.75. The length of arrangement, 38.95			
1	2	4	3
1	2	1	1
93	64	70	84
Where λ_i×2 for i=1,…,m. The fitness function, 92.54. The length of arrangement, 26.85			
1	3	4	
1	1	1	
86	77	64	
Where λ_i×0.8 for i=1,…,m. The fitness function, 44.13. The length of arrangement, 24.41			
1	2	4	3
1	2	1	1
31	21	23	28
Where λ_i×0.5 for i=1,…,m. The fitness function, 26.72. The length of arrangement, 16.71			
1	2	3	4
1	2	1	1
17	12	16	13

5.3 The effect of the changeover times and costs on the solutions

The changeover times and costs have an important effect on the solutions. So, if these values approach zero, the optimum batch sizes approach smaller values as much as possible. Contrarily, if these values are very high, the optimum number of runs may be as small as possible to avoid additional changeover times and costs. In this situation, the optimum batch sizes are determined through

Table 9 The proposed solutions by GA where only λ_1 is multiplied by 2 and 0.5

Task 1	Task 2	Task 3	Task 4	Task 5
Where λ_1×2. The fitness function, 64.1. The length of arrangement, 39.9				
1	2	4	1	3
1	2	1	1	1
74	43	47	53	57
Where λ_1×0.5. The fitness function, 50.2. The length of arrangement, 25.26				
1	2	3	4	
1	2	1	1	
20	27	30	36	

Table 10 The proposed solutions by GA where all of production rates are multiplied by 2, 5, 0.8, and 0.5

Task 1	Task 2	Task 3	Task 4	Task 5
Where $D_i\times2$ for $i=1,\ldots,m$. The fitness function, 56.63. The length of arrangement, 21.09				
1	2	4	3	
1	2	1	1	
33	23	25	30	
Where $D_i\times5$ for $i=1,\ldots,m$. The fitness function, 56.89. The length of arrangement, 22.9				
1	2	4	1	3
1	2	1	1	1
20	25	27	16	33
Where $D_i\times0.8$ for $i=1,\ldots,m$. The fitness function, 53.53. The length of arrangement, 32.08				
1	2	4	3	
1	2	1	1	
51	35	38	46	
Where $D_i\times0.5$ for $i=1,\ldots,m$. The fitness function, 45.16. The length of arrangement, 26.57				
1	3	4		
1	1	1		
42	38	31		

a compromise between inventory holding costs and changeover costs. In other words, the bigger batch sizes increase the inventory holding costs, but decrease the portion of changeover costs in the total production costs. In some cases, if these parameters are very high, GA may remove some product families from the optimum arrangement to avoid the related changeover times and costs. Some results of GA for different values of these parameters are shown in Table 11.

Where these parameters approach zero, the number of the multiple optimum solutions is increased and GA hardly converges to a specific solution, but the fitness functions of different solutions get very near to each other.

Table 11 The proposed solutions by GA where all of changeover times are multiplied by 2 and 0.2

Task 1	Task 2	Task 3	Task 4
Where $GT(c_{i,j}, c_{k,l})\times2$ for all i, j, k, and l. The fitness function, 42.19. The length of arrangement, 54.78			
1	2	4	3
1	2	1	1
87	60	65	79
Where $GT(c_{i,j}, c_{k,l})\times0.2$ for all i, j, k, and l. The fitness function, 60.97. The length of arrangement, 5.26			
1	2	4	3
1	2	1	1
8	5	6	7

Table 12 The proposed solutions by GA where all of inventory holding costs are multiplied by 3 and 0.2

Task 1	Task 2	Task 3	Task 4
Where $h_i\times3$ for $i=1,\ldots,m$. The fitness function, 35.94. The length of arrangement, 16.32			
1	3	4	
1	2	1	
26	23	19	
Where $h_i\times0.2$ for $i=1,\ldots,m$. The fitness function, 65.02. The length of arrangement, 30.47			
1	3	2	4
1	1	2	1
48	43	33	36

5.4 The effect of the coefficient of inventory holding costs on the solutions

In general, if the coefficients of inventory holding cost (h_i) increase, the smaller batch sizes may be preferred. The violence of this effect is low because of the importance of other parameters such as the production rates and the arrival rates of orders. In this example, if all of the inventory costs are multiplied by 2, the optimum arrangement and its optimum batch sizes are not changed in comparison with the primitive solutions; but if they are multiplied by 3, the product family 2 is removed from the optimum arrangement and smaller batch sizes for the remaining product families result.

Similarly, if the inventory holding cost for a product family increase a little, the optimum solutions do not change. But, if it increases further, this product family may be repeated in the optimum arrangement with smaller batch sizes. For its greater value, this product may be removed from the optimum

Table 13 The proposed solutions by GA where only h_4 is multiplied by 3 and 20

Task 1	Task 2	Task 3	Task 4	Task 5
Where $h_4\times3$. The fitness function, 42.21. The length of arrangement, 16.32				
1	2	4	3	4
1	2	1	1	1
48	33	14	44	22
Where $h_4\times20$. The fitness function, 39.51. The length of arrangement, 22.29				
1	2	3		
1	2	1		
35	24	32		

Table 14 The parameters and characteristic of new configuration of product family 1

Production configuration cost $(CC_{1,3})=18$	Production configuration rate $(CR_{1,3})=15$			
Changeover cost: $GC(C_{i,j},C_{k,l})$	$GC(C_{1,3},C_{1,1})=11.0$	$GC(C_{1,3},C_{2,1})=12.7$	$GC(C_{1,3},C_{3,1})=11.8$	$GC(C_{1,3},C_{4,1})=12.5$
	$GC(C_{1,3},C_{1,2})=10.7$	$GC(C_{1,3},C_{2,2})=12.1$	$GC(C_{1,3},C_{3,2})=10.0$	$GC(C_{1,3},C_{4,2})=11.7$
	$GC(C_{1,1},C_{1,3})=12.0$	$GC(C_{1,3},C_{2,3})=11.3$	$GC(C_{3,1},C_{1,3})=12.0$	$GC(C_{4,1},C_{1,3})=10.5$
	$GC(C_{1,2},C_{1,3})=9.2$	$GC(C_{2,1},C_{1,3})=13.1$	$GC(C_{3,2},C_{1,3})=11.7$	$GC(C_{4,2},C_{1,3})=10.8$
		$GC(C_{2,2},C_{1,3})=12.5$		
		$GC(C_{2,3},C_{1,3})=12.0$		
Changeover time: $GT(C_{i,j},C_{k,l})$	$GT(C_{1,3},C_{1,1})=1.1$	$GT(C_{1,3},C_{2,1})=3.7$	$GT(C_{1,3},C_{3,1})=2.8$	$GT(C_{1,3},C_{4,1})=2.5$
	$GT(C_{1,3},C_{1,2})=2.7$	$GT(C_{1,3},C_{2,2})=3.1$	$GT(C_{1,3},C_{3,2})=2.0$	$GT(C_{1,3},C_{4,2})=3.7$
	$GT(C_{1,1},C_{1,3})=1.0$	$GT(C_{1,3},C_{2,3})=2.3$	$GT(C_{3,1},C_{1,3})=3.1$	$GT(C_{4,1},C_{1,3})=2.5$
	$GT(C_{1,2},C_{1,3})=1.9$	$GT(C_{2,1},C_{1,3})=2.1$	$GT(C_{3,2},C_{1,3})=3.7$	$GT(C_{4,2},C_{1,3})=2.8$
		$GT(C_{2,2},C_{1,3})=2.5$		
		$GT(C_{2,3},C_{1,3})=3.0$		

arrangement. The solutions in Tables 12 and 13 are determined by GA where all of the holding costs are multiplied by 3 and 0.2, and where the inventory holding cost for product family 4 is multiplied by 2 and 3, respectively.

5.5 The effect of introducing a new configuration on the solutions

Suppose that $c_{1,3}$ is a new designed configuration for product family 1. Its main parameters and characteristics are shown in Table 14.

Considering the new configuration, GA converges to the solution in Table 15.

The fitness functions and the length of corresponding arrangement to the converged solution are 59.43 and 22.7, respectively. The new configuration improves the fitness function about \$4.63/min. Suppose that designing and constructing this configuration have a cost of \$100,000. Thus, the new system can cover this cost during 360 $(100{,}000/(4.63\times60))$ working hours.

5.6 The effect of introducing a new product family on the solutions

Suppose that a new product family having three feasible configurations is designed to cover new market demand. The main parameters and characteristics of new product family and its feasible configurations are shown in Table 16.

Considering the new product family, GA converges to solution shown in Table 17.

The fitness functions and the length of the converged solution are 77.98 and 32.85, respectively. The product family improves the fitness function about \$23.18/min. Suppose that initial investment to run the new product is about \$1,500,000. Thus, the new system can cover this cost during 1,078.5 working hours.

6 Discussions

The solution procedure is coded by C++. Numerous numerical experiments are conducted to illustrate the effectiveness of the proposed model and its solution procedure. The numeration method and the sensitive analyzing have shown the accuracy of the results of the proposed solution procedure. The proposed GA procedure often converges to the optimum solutions within 50 generations. Using a Pentium IV personal computer, the GA solution procedure converges in less than 1 min while numeration method may take 1 h.

This methodology can be applied to evaluate the performance criteria of an RMS. The alternative scenarios that may improve the performance criteria of the RMS can be evaluated by proposed GA model. For example, where the increase on the arrival rates of product families improves the objective functions of the model, it may be concluded that the production capability of the RMS is higher than the current market demand and it can handle a higher rates of orders. It may be also concluded that the RMS has external constraints such as marketing activities and should be considered as the first priority to improve the performance of the RMS. Vice versa, if the increase on the orders' rates does not improve the optimum solutions and also increases the missed orders, it may imply that the RMS cannot handle a higher rate of orders. It may be concluded

Table 15 The proposed solutions by GA where a new configuration is introduced for product family 1

Task 1	Task 2	Task 3	Task 4	Task 5
1	2	4	1	3
3	2	1	3	1
22	24	27	14	32

Table 16 The parameters and characteristics of new product family

Arrival rate λ_5=2.1	Selling price S_5=30		Inventory holding cost h_5=0.25		
Feasible production configurations $C_5=\{c_{5,1}, c_{5,2}, c_{5,3}\}$					
	Production configuration cost={19, 21,23}		Production configuration rate={10, 12,14}		
Changeover cost: GC($C_{i,j}$,$C_{k,l}$)	GC($C_{5,j}$,$C_{1,l}$)=	GC($C_{5,j}$,$C_{2,l}$)=	GC($C_{5,j}$,$C_{3,l}$)=	GC($C_{5,j}$,$C_{4,l}$)=	GC($C_{5,j}$,$C_{5,l}$)=
	15.1 14.6	16.0 15.7 14.7	12.3 11.7	14.2 14.7	0.0 10.1 11.8
	14.1 14.7	16.0 15.7 14.6	14.2 12.8	13.0 15.7	13.0 0.0 12.1
	15.1 12.7	13.0 14.7 13.8	13.1 15.7	12.9 14.7	12.9 13.1 0.0
	GC($C_{1,l}$,$C_{5,j}$)=	GC($C_{2,l}$,$C_{5,j}$)=	GC($C_{3,l}$,$C_{5,j}$)=	GC($C_{4,l}$,$C_{5,j}$)=	
	13.0 13.2 14.0	12.0 14.1 13.2	14.3 12.3 15.1	14.1 15.2 13.1	
	13.7 12.7 15.7	12.2 13.1 14.7	16.7 16.1 15.8	12.7 13.9 12.3	
		13.8 15.7 15.6			
Changeover time: GT($C_{i,j}$,$C_{k,l}$)	GT($C_{5,j}$,$C_{1,l}$)=	GT($C_{5,j}$,$C_{2,l}$)=	GT($C_{5,j}$,$C_{3,l}$)=	GT($C_{5,j}$,$C_{4,l}$)=	GT($C_{5,j}$,$C_{5,l}$)=
	5.1 4.6	6.0 5.7 4.7	2.3 1.7	4.2 4.7	0.0 1.1 1.8
	4.1 4.7	6.0 5.7 4.6	4.2 2.8	3.0 5.7	3.0 0.0 2.1
	5.1 2.7	3.0 4.7 3.8	3.1 5.7	2.9 4.7	2.9 3.1 0.0
	GT($C_{1,l}$,$C_{5,j}$)=	GT($C_{2,l}$,$C_{5,j}$)=	GT($C_{3,l}$,$C_{5,j}$)=	GT($C_{4,l}$,$C_{5,j}$)=	
	3.0 3.2 4.0	2.0 4.1 3.2	4.3 2.3 5.1	4.1 5.2 3.1	
	3.7 2.7 5.7	2.2 3.1 4.7	6.7 6.1 5.8	2.7 3.9 2.3	
		3.8 5.7 5.6			

that there are some internal constraints. These internal constraints should be released as soon as possible. In such situation, the production capacity may be increased by the introduction of a new configuration. This modification improves the production rate of corresponding product family or reduces changeover times.

As a part of numerical experiences, the effects of inventory holding costs on optimum arrangement were studied. The higher inventory holding costs will result to the smaller batch sizes. The results show that, if changeover cost and time approach zero, then optimum batch sizes reduces. Conversely, if changeover times and costs are increased, then the batch sizes at each production task are increased.

7 Conclusion

In the new business era embracing "change" as one of its major characteristics, the manufacturers need to adapt to approaches that can lead them to achieve more adaptability to market changes. RMS is a system designed, from the outset, for rapid changes in structure, both in hardware and software components, in order to quickly adjust production capacity and functionality within a part family in response to sudden market changes.

Table 17 The proposed solutions by GA where a new product family is introduced to RMS

Task 1	Task 2	Task 3	Task 4	Task 5
1	2	4	5	3
1	2	1	3	1
52	36	39	32	47

The effectiveness of an RMS depends on implementing its key characteristics and principles in the design and utilization stage. This paper focuses on the utilization stage of an RMS and introduces a mathematical model to manage and evaluate effectiveness of RMS. This model considers the key characteristics and capabilities of RMS to adjust scalable production capacities and the functionality of the system to respond rapidly to market demands and fulfill productivity. There are many uncertain factors affecting the operation performance of an RMS; therefore, stochastic parameters are concerned for modeling purposes. This paper concerns the following issues in an RMS mathematical modeling:

- Optimum sequence of selecting product families as production tasks.
- Optimum configuration of the selected product family.
- Optimum batch sizes.

An arrangement defines the sequence of a finite number of production tasks that are repeated over a time horizon. To use stochastic parameters in mathematical models, the on-hand inventory levels, inventory holding costs, and sales are evaluated and estimated according to stochastic orders. These estimated values are used in a MINLP model. The goal of the model is to maximize the rate of profit. This rate is defined as selling prices minus production costs divided by the arrangement time.

The solution comprised of the following procedures: (1) set the number of the production tasks within an arrangement, (2) get the sequence of production tasks from genetic algorithm model; production tasks are comprised of product families and their configurations, and (3) determine the optimum batch sizes and the fitness value of proposed arrangement by GA.

Advantages of the proposed model and its optimization procedure are summarized as follows:

- The model assumes that the manufacturer should respond to arrival orders immediately. Therefore, the model focuses on minimizing the inventory holding costs and production costs where response time is set to zero. This is a suitable assumption in a highly competitive market environment.
- In comparison with Zaho et al. [4], the proposed model suggests all feasible configurations in utilization stage. It also helps designers to determine the best design parameters of a new configuration.
- In comparison with Abbasi and Houshmand's model [12], the proposed model suggests a precise solution and determines reasonable batch sizes. The suggested batch sizes ensure the best coverage of arrival orders with minimum production costs. Moreover, in the integration of all solution procedures in a C++ program, the optimum solutions are determined automatically in a very short time.
- In this research, a numeration method is applied to validate the results. This method is very time consuming, but it determines global optimum solutions. Comparison of the numeration and gradient method shows that gradient method achieves the optimum solution in less iteration.
- Using this modeling approach and the proposed solution procedure, the effects of alternative decisions on the RMS performance can be evaluated. This method can be considered as a decision support system to evaluate a new product family, a new configuration, a new marketing policy, etc.

The following issues can be seen as future directions of this research:

- The proposed model can be applied to evaluate different scenarios of grouping products into families [21]. In the other hand, this model can help the part manufacturer to determine the best manufacturing system to produce a product family [22].
- In some businesses environments, it is preferable to respond to delayed orders as soon as they get ready. Implementing the shortage penalty for delayed orders may be proposed as a further research.
- Since the arrival orders are stochastic, the current condition of RMS is changed unpredictably. Therefore, manufacturers need a dynamic production scheduler that makes decisions upon current situation of an RMS. This research can be considered as a base to develop such scheduler.
- Since there is no practical RMS, a numerical example is used to validate the results of the proposed model and its solution procedure. However, the authors are working on an OEM's assembly line whose principles and characteristics are near to actual RMS'. The results and achievements of this research will be presented in future works.

References

1. Koren Y (2006) General RMS characteristics. Comparison with dedicated and flexible systems. Chapter 3 in the reconfigurable manufacturing systems and transformable factories. Springer, Netherlands
2. Mehrabi MG, Ulsoy AG, Koren Y (2000) Reconfigurable manufacturing systems: key to future manufacturing. J Intell Manuf 11:403–419
3. Chick SE, Olsen TL, Sethuraman K, Stecke KE, White CC (2000) A descriptive multi-attribute model for reconfigurable machining system selection examining buyer–supplier relationships. Int J Agil Manag Syst 2(1):33–48
4. Zhao X, Wang J, Zhenbi L (2000) A stochastic model of a reconfigurable manufacturing system, part 1: a frametask. Int J Prod Res 38(10):2273–2285
5. Zhao X, Wang J, Zhenbi L (2000) A stochastic model of a reconfigurable manufacturing system, part 2: optimal configurations. Int J Prod Res 38(12):2829–2842
6. Zhao X, Wang J, Zhenbi L (2001) A stochastic model of a reconfigurable manufacturing system, part 3: optimal selection policy. Int J Prod Res 39(4):747–758
7. Zhao X, Wang J, Zhenbi L (2001) A stochastic model of a reconfigurable manufacturing system, part 4: performance measure. Int J Prod Res 39(6):1113–1126
8. Yang T, Peters BA (1998) Flexible machine layout design for dynamic and uncertain production environments. Eur J Oper Res 108:49–64
9. Kochhar JS, Hwragu SS (1999) Facility layout design in a changing environment. Int J Prod Res 37:2429–2446
10. Lee GH (1997) Reconfigurability consideration design of components and manufacturing systems. Int J Adv Manuf Technol 13:376–386
11. Takahashi K, Morikawa K, Myreshka, Ohiro T, Takuba A (2006) A stochastic model for deciding an optimal production order and its corresponding configuration in a reconfigurable manufacturing system with multiple product groups. Chapter 30 in the reconfigurable manufacturing systems and transformable factories. Springer, Netherlands
12. Abbasi M, Houshmand M (2009) Production planning of reconfigurable manufacturing systems with stochastic demands using tabu search. Int J Manuf Technol Manage 17(1/2):125–148

13. Gross D, Harris CM (1985) Fundamentals of queuing theory. Wiley, New York
14. Holland JH (1975) Adaptation in natural and artificial systems. The University of Michigan Press, Ann Arbor
15. Gen M, Cheng R (2000) Genetic algorithms and engineering optimizations. Wiley, New York
16. Tay JC, Wibowo D (2004) An effective chromosome representation for evolving flexible job shop schedules. GECCO. In: Lecture Notes in Computer Science, vol. 3103. Springer, Berlin, pp. 210–21
17. Pezzella F, Morganti G, Ciaschetti G (2007) A genetic algorithm for the flexible job-shop scheduling problem. Comput Oper Res 35:3202–3212. doi:10.1016/j.cor.2007.02.014
18. Iyer SK, Saxena B (2004) Improved genetic algorithm for the permutation flow shop scheduling problem. Comput Oper Res 31:593–606
19. Liaw CF (2000) A hybrid genetic algorithm for the open shop scheduling problem. Eur J Oper Res 124:28–42
20. Chu PC, Beasley JE (1997) A genetic algorithm for the generalized assignment problem. Comput Oper Res 24(1):17–23
21. Abdi MR, Labib AW (2004) Grouping and selecting products: the design key of reconfigurable manufacturing systems (RMSs). Int J Prod Res 42(3):521–546
22. Abdi M, Labib AW (2003) A design strategy for reconfigurable manufacturing systems (RMSs) using analytical hierarchical process (AHP): a case study. Int J Prod Res 41(10):2273–2299