

# A GENETIC ALGORITHM FOR A SIMULTANEOUS OPTIMISATION OF COST-RISK REDUCTION UNDER A JUST-IN-TIME ADAPTION

<sup>1,2</sup>Faraj El Dabee, <sup>1</sup>Romeo Marian and <sup>1</sup>Yousef Amer

<sup>1</sup>School of Engineering, University of South Australia, Adelaide, Australia

<sup>2</sup>Department of Mechanical Engineering, Faculty of Engineering, University of Elmergib, Al-Khoms, Libya

Received 2014-09-03; Revised 2014-11-13; Accepted 2014-11-25

## ABSTRACT

Just-In-Time (JIT) as a lean manufacturing approach plays a significant role in minimising costs and performances of products and services supplied to the global marketplace. However, there are many potential risks that cause significant disruptions to all supply chain members. This study proposes a genetic approach for optimising a novel mathematical model for simultaneously minimising the total cost of a final product and the potential risks related to these benefits. Specifically, it demonstrates the effectiveness of a genetic algorithm in optimising the JIT model developed in our previous paper. Genetic operators adopted to improve the genetic search algorithm are introduced and discussed. Experiments are carried out to evaluate the performance of the proposed algorithm using a simplified example. Comparison of four selection methods is done to define the best method that can be used in the proposed GA. The findings demonstrate the superiority of the proposed approach in the JIT system with focus on simultaneous cost-risk reduction.

**Keywords:** Just-In-Time (JIT), Production System, Cost-Risk Reduction, Model, Optimisation, Genetic Algorithm (GA)

## 1. INTRODUCTION

Optimisation problems arise in case discrete choices must be made and solving them amounts to find an optimal solution among a large number of alternatives. Recently, researchers have proposed many approaches that can be used for obtaining the optimum solutions within a reasonable amount of time (Tasan and Tunali, 2008). Recently, many meta-heuristics optimisation approaches have been applied in the supply chain management field to solve engineering optimisation issues (Wang and Wang, 2008). These approaches are associated with transportation/distribution networks consider the location of the organisation, design of the network configuration and customer satisfaction by

minimising the total product cost (Paksoy and Chang, 2010). Optimisation is the process of adapting the inputs of a device and mathematical process to find the minimum or maximum of the output. Optimisation is a main technique that can be used for addressing complex problems. The key purpose of optimisation is to find the global optima (maximum or minimum) of a formulated objective function to a problem. Optimisation problems are used for acquiring good component parameters to be set into activities by humans or machines (Malhotra *et al.*, 2011).

Numerous industrial engineering design problems are very complex and hence intractable for conventional optimisation techniques. Evolutionary Algorithms (EAs) are population-based meta-heuristic optimisation

**Corresponding Author:** Faraj El Dabee, School of Engineering, University of South Australia, Australia

algorithms that refine a set of solution iteratively using biology-inspired mechanisms and theory of the survival fittest (Malhotra *et al.*, 2011). The evolution process initiates from a population of generated individuals and occurs in generations (Kannaiah *et al.*, 2011).

Globally, many organisations have used a JIT approach in their processes to achieve their objectives. However, some have ignored certain significant risks which arise from its implementation such as natural disasters. Businesses are exposed to numerous risks, particularly those originating from the supply chain. These risks potentially influence their processes by disrupting all supply chain members involved (El Dabee *et al.*, 2013b; 2013c). In this context, risk is defined as the combination of possible consequences and associated uncertainties (Aven and Vinnem, 2007). Generally, supply chain risk is an event exposure that causes disruption affecting the efficient processes of the whole supply chain (Ghadge *et al.*, 2012).

The main objective of this study is to develop an optimisation Approach (GA) for the proposed JIT model published in (El Dabee *et al.*, 2013a). The novelty of this study is the simultaneous cost-risk reduction of the final product in the production system under a JIT environment by using the GA approach for finding the optimum solution of the proposed model.

In the remainder of this study, Subsection 1.1 reviews the literature on JIT approach using optimisation techniques and Subsection 1.2% an overview of the JIT model reported in our previous study in (El Dabee *et al.*, 2013a); section 2 presents the materials and methods adopted in this study includes the proposed GA parameters for optimising the JIT model and implementing the proposed GA in a simplified example to find the optimal or near-optimal solution of cost-risk reduction in JIT systems; section 3 discusses and analyses the obtained results to illustrates the effectiveness of the proposed GA; while section 4 summarises the paper.

### 1.1. An Overview of JIT Optimisation Using GA

In today's increasingly competitive environment, the adoption of efficient approaches plays a decisive role in successful performance of organisations (Cai *et al.*, 2009). This can be achieved by the continuous improvement and optimisation of processes and operations, cost reduction of services and products and an increased outputs capacity with satisfactory product quality and production rates (El Dabee and Hokoma,

2012; Hokoma *et al.*, 2010). JIT as one of the most significant approaches can be used for eliminating all waste and non-value added activities within organisations, in order to minimise products cost (Krajewski *et al.*, 2013).

Recently, GA as an evolutionary or met-heuristic algorithm has received a substantial attention as robust optimisation techniques to find the optimal or near-optimal solution within a reasonable amount of time (Geem *et al.*, 2005; Sivanandam and Deepa, 2008; Yussof *et al.*, 2011). GA is a subclass of Evolutionary Algorithms (EAs) where the elements of the search space are strings or arrays of other elementary types (Rawal and Inamdar, 2014). Based on their simplicity, ease of operation, minimal requirements and parallel and global perspective, GA has been widely applied in a variety of problems (Sivanandam and Deepa, 2008). GA can optimise with discrete, continuous and mixed parameters cope with a large number of parameters; and do not require derivative information (Marian, 2003; Park *et al.*, 2011).

A JIT distribution model for three-level supply chain network was developed to minimise the product cost, the sum of backorders and product surpluses during all stages in JIT distribution systems. The model considers distribution lead time and capability restrictions in multiple periods, products and channel networks. A HGA was designed to solve real large-size problems of the proposed model. The results obtained from some small-size test problems were compared with the results obtained by LINGO optimisation software (Farahani and Elahipanah, 2008). GA and fuzzy set approach was applied to generate JIT scheduling to optimise the fabric-cutting process in the apparel industry. Data was collected from fabric-cutting department of Hong Kong-owned apparel manufacturing company to validate the proposed optimisation method. The findings illustrated the effectiveness of this approach, which improve the performance and minimise the production cost within the organisation (Wong *et al.*, 2006).

Ghasimi *et al.* (2014) introduced a novel mathematical model for three-echelon Defective Goods Supply Chain Network (DGSCN) was built to ascertain the Economic Production Quantity (EPQ) and lead time, additionally reducing all production, distribution and holding cost using JIT logistics. GA and Cplex solver were used to optimise their defective goods supply chain network model. The model outputs were compared to define its performance and find the proper Length for Each Cycle (ALOEC). Jianhua and Xianfeng (2010)

proposed a Hybridised Genetic Algorithm (HGA) by merging GA with the Greedy Algorithm (GrA) to solve the Agile Supply Chain Scheduling (ASCS) problem in order to reduce the inventory and transportation cost in the supply chain. The authors showed that practical production and transportation schedules ensure the implementation of JIT while reducing the costs of operation in the supply chain.

A Multi-Objective Genetic Algorithm (MOGA) approach in JIT sequencing problem environment was proposed. MOGA was used to simultaneously minimise the number of required setups and the production rate variation. Extensive computational experiments were applied on the proposed algorithm to demonstrate its effectiveness. The results showed that this approach has a significant impact on the quality improvement compared with Total Enumeration (TE) approach in a short time (Mansouri, 2005).

According to the aforementioned literature related to a GA optimisation, there is no research to date addresses the simultaneous cost-risk reduction in the supply chain, particularly, JIT systems.

## 1.2. The Proposed Model

This Subsection presents the main optimisation function of the proposed model developed for simultaneously minimising the cost of the final product and the risks effect in a production system under JIT environment in (El Dabee *et al.*, 2013a). This model incorporates four main parties, which are supplier, manufacturer, distribution centre and end user. It is assumed that a distribution network consists of multiple external suppliers who supply raw materials or components/subassemblies to the production system to produce the final product. This assumption is based on the pricing variations for the same product in different markets. The materials are transported from different manufacturers to the production system, which in turn produce the final product for sale to wholesale or retail outlets. Also, the raw materials are supplied instantaneously to the production system to meet JIT requirements. The products are delivered to the end customers such as wholesalers or retailers with no holding capacity to store the products.

Many risks may result from unforeseen disruptions such as natural and man-made disasters and economic crises affecting external suppliers. All of them have a significant impact on the production facility and the entire supply chain as well. To avoid the impact of these risks, it is assumed that during a time of

disruption, the production system can procure its raw materials or components/subassemblies required to produce the final product from a local backup supplier at a higher price but with low risk and in a short lead time (El Dabee *et al.*, 2013b).

The model also considers scenarios in case orders for raw materials are shipped by both suppliers using different transportation modes: Waterways, railways, roadways and the airways (Murphy and Wood, 2011). The appropriate transportation mode can be selected depending on some key criteria such as the shortest time and the transportation cost required for carrying raw materials to the production system. The model is coded to ascertain the total cost of producing the final product within production systems.

The following notations are used in the proposed model:

- $C_T$ : Total cost required to produce one product in Monetary Unit (MU)
- $C_{pt}$ : Final product cost excluding the risk cost in Monetary Unit (MU)
- $C_{RM}$ : Raw material cost required for producing one product (MU)
- $C_O$ : Ordering cost of raw Materials (MU)
- $C_H$ : Holding cost of raw materials within the production system stores (MU)
- $C_R$ : Risk cost arising from disruption occurrence (MU)
- $C_{ir}$ : Transportation cost for delivering raw materials to the production system (MU)
- $C_P$ : Purchasing cost of raw materials required to produce the product (MU)
- $C_U$ : Utilities cost of the final product (MU)
- $C_D$ : Duties cost arising from procuring raw material from an external supplier (MU)
- $TP_i$ : Transfer price required for procuring raw material  $i$  from an external supplier  $i$  (MU)
- $S$ : Origin of ordered raw materials
- $v$ : Destination of required raw materials
- $m_i$ : Transportation mode for transporting raw material  $i$  to its customer
- $N_T$ : Number of transportation modes used for shipping raw materials to the production system
- $S_{Ej}$ : Raw material external supplier  $j$
- $S_{LBS}$ : Raw material local backup supplier  $s$
- $IF$ : Indicator function for duty with a value 1 or 0.1 if the supplier and the production facility are in the same country and 0 otherwise

- *LH*: Likelihood of occurrence for risk in the supply chain
- *I*: Impact of risk occurrence in the supply chain
- *N<sub>O</sub>*: Number of operations required for producing one product (unit)
- *N<sub>p</sub>*: Number of parts required to produce one product
- *N<sub>SLB</sub>*: Number of local suppliers used for supplying raw materials to the production system (unit)
- *C<sub>UO</sub>*: Ordering cost of raw materials for the final product (MU)
- *C<sub>UH</sub>*: Holding cost of raw materials of each final product in the production system warehouse (MU/day)
- *C<sub>UMi</sub>*: Unit cost of the raw material *i* at the beginning of each cycle (MU)
- *%d<sub>RM</sub>*: Daily demand percentage of raw materials required to meet customer satisfaction (unit)
- *h*: Operation time required to produce a product (hr)
- *C<sub>L</sub>*: Labour cost rate per time in one operation (MU/hr)
- *SF*: Storage factor for keeping raw materials in the warehouse
- *OF*: Ordering factor for procuring each order from the supplier
- *%Util*: Utilities cost percentage of the final product
- *t<sub>p</sub>*: Raw material cost percentage incurred for procuring raw material *i* from an external supplier *j*
- *LT*: Lead-time taken between placing and receiving the placed order (day)
- *T<sub>Sj v, m</sub>*: Tensor for transportation cost per critical measurement to transport raw materials from its origin *s* to its destination *v* using transportation mode *m* (MU)
- *%V*: Volume percentage value required for transporting raw materials to their customer
- *D<sub>i</sub>*: Duty rate (%) per price of raw material *i* supplied by external supplier *j* (MU); and
- *%TRS* Total risk score percentage value

The purpose of this model is to optimise the total costs (*C<sub>T</sub>*), including *C<sub>O</sub>*, *C<sub>H</sub>*, *C<sub>p</sub>*, *C<sub>ir</sub>*, *C<sub>D</sub>*, *TP*, *C<sub>w</sub>*, *C<sub>U</sub>* and *C<sub>R</sub>* using the GA technique. The total cost of this product can be calculated as Equation 1 and 2:

$$C_T = C_{pt} + C_R \tag{1}$$

$$C_{pt} = C_O + C_H + C_p + C_{ir} + C_D + TP + C_w + C_U \tag{2}$$

*C<sub>T</sub>* can be calculated in the case of using the external supplier for procuring raw materials as follows Equation 3:

$$C_T = \sum_{j=1}^{N_{SE}} C_{UOj} \times OF + \sum_{i=1}^{N_p} \sum_{j=1}^{N_{SE}} (C_{UH_i})_j \times \%d_{RM} \times (LT_j + SF) + \sum_{i=1}^{N_p} \sum_{j=1}^{N_{SE}} (C_{UMSE_i})_j + \sum_{j=1}^{N_{SE}} \sum_{l=1}^{N_T} \sum_{i=1}^{N_p} T_{SLB_j, v, m} \times t_{m_i} \times \%V_i + \sum_{i=1}^{N_p} \sum_{j=1}^{N_{SE}} C_{UP_i} (1 - IF_j) \times D_j + \sum_{j=1}^{N_{SE}} \sum_{i=1}^{N_p} TP_j \times C_{UP_i} + \sum_{i=1}^{N_p} C_{L_i} \times h_i + \sum_{i=1}^{N_p} \%util \times C_{RM_i} + \sum_{i=1}^{N_p} \sum_{k=1}^{LH_k} \frac{LH_k \times I_k}{Max(LH_k \times I_k)} \times C_{pt_i} \tag{3}$$

Also, when raw materials are supplied by the local backup supplier, *C<sub>T</sub>* can be found as Equation 4:

$$C_T = \sum_{s=1}^{N_{SLB}} C_{UOs} \times OF + \sum_{i=1}^{N_p} \sum_{s=1}^{N_{SLB}} (C_{UH_i})_s \times \%d_{RM} \times (LT_j + SF) + \sum_{i=1}^{N_p} \sum_{s=1}^{N_{SLB}} (C_{UMSLB_i})_s + \sum_{s=1}^{N_{SLB}} \sum_{l=1}^{N_T} \sum_{i=1}^{N_p} T_{SLB_s, v, m} \times t_{m_i} \times \%V_i + \sum_{i=1}^{N_O} C_{L_i} \times h_i + \sum_{i=1}^{N_p} \%util \times C_{RM_i} + \sum_{i=1}^{N_p} \sum_{k=1}^{LH_k} \frac{LH_k \times I_k}{Max(LH_k \times I_k)} \times C_{pt_i} \tag{4}$$

## 2. MATERIALS AND METHODS

GA is a modern heuristic algorithm, which is widely adopted by researchers in solving complex and large-scale combinatorial optimisation problems (Littidej and Suttayamully, 2009; Radhakrishnan *et al.*, 2009; Ali *et al.*, 2010). GA is an adaptive heuristic based on ideas of natural selection and genetics and one of the most popular approaches of evolutionary algorithms (Gupta and Ghafir, 2012). GA is the key paradigm of evolutionary computation, which are inspired by Darwin's theory of the survival of the fittest (Scavino *et al.*, 2009; Zhu and Kwong, 2010). GA retains a population of different solution allowing them to mate, produce offspring, mutate and fight for survival. The principle of survival of the fittest ensures the populations derive towards optimisation (Wang and Wang, 2008).

The proposed GA was coded using Java language and run on a PC Intel® Core™ i5-3210M processor working at speed 2.5 GHz and RAM of 4GB. This study uses GAs to study the simultaneous cost-risk reduction under the JIT approach to find the optimal solution of the developed model used for this purpose, which has been tested with a simplified example. The

key components of GA are: The representation of the solution, creation of initial population, evaluation of solutions, the development of genetic operators (crossover and mutation) and the termination creation (Go *et al.*, 2010; Goren *et al.*, 2010). Also, Mahmudy *et al.* (2012) summarise the main stages of GA as the key parameters setting, initialisation, reproduction and selection stages. All these stages will be described briefly as follows.

### 2.1. Parameters Setting

Setting GA parameters is the first stage that includes the population size ( $N$ ), crossover rate ( $P_C$ ), mutation rate ( $P_m$ ), maximum Generations ( $G_{max}$ ) and the number of generations after which the GA will stop if no further improvement for the last best solution ( $G_{max\_Stop}$ ) is achieved. Based on these parameters, the population size can be created easily.

### 2.2. Initial Population Generation

GA starts with generation of a number of solutions (chromosomes), called a population. The population is usually randomly initialized (Al-Taharwa *et al.*, 2008; Kumar *et al.*, 2012). This stage is a key issue in GA approach based on their significant impacts on the subsequent stages. Selecting a good initial population with an appropriate population size can greatly improve the GA efficiency (Torabi *et al.*, 2006). Each chromosome/individual is constituted by genes which are generated with random values. Chromosomes evolve through successive iterations, called generations.

#### 2.2.1. Chromosome Representation

The first and one of the most significant steps in applying GA to a particular problem is to convert solutions (chromosomes) of a JIT cost-risk reduction problem into a string type structure called chromosome. Each chromosome represents one potential solution. The chromosome is a string of symbols that can be coded in different forms such as binary, integer and real numbers (Gen and Cheng 2000). The chromosome refers to the random population of encoded candidate solutions with which the Genetic algorithms initiate with (Radhakrishnan *et al.*, 2009). This study uses a real number representation, called Real-Coded Genetic Algorithms (RCGA). RCGA considers each chromosome as a vector of real numbers. For a problem of  $n$  variables, the structural representation of a chromosome,  $X$  is:  $X = (X_1 + X_2 + X_3, \dots, X_n)$  (Gen and Cheng, 2000). Here, the chromosome of 4 genomes ( $X_1$ ,

$X_2$ ,  $X_3$  and  $X_4$ ) represents the main decision variables with 34 genes.  $X_1$  presents the number of external suppliers ( $S_{Ej}$ ) used for supplying raw materials to the production system, which are 11 suppliers (genes) with 2 levels. if  $X_1 = 1$ ,  $S_E$  can supply raw materials and  $C_T$  is calculated using Equation 3 and if  $X_1 = 0$ ,  $S_E$  has disruption and  $C_T$  is calculated using Equation 4;  $X_2$  presents the customer demand ( $d_p$ ) for the final product per day with 1 demand (gene), which has 210 levels ( $d_p = 1, 2, \dots, 210$ );  $X_3$  presents the critical transportation measurement ( $t_m$ ) of raw materials shipped using transportation mode  $m$  with 11 suppliers (genes), which has 4 levels ( $m = 1, 2, 3, 4$ ); finally,  $X_4$  presents the quantity of raw material ( $Q_M$ ) ordered in each patch required to produce the final product per week using 11 suppliers (genes) with 7 levels ( $Q_M = 350, 700, 1050, 1400, 1750, 2100, 2450$ ). A random individual generated for the genetic operation is shown in **Table 1**.

By generating the individuals, the number of occurrences of the individual in the previous records is determined. Based on the four decision variables used, the number of solutions can be calculated as  $2^{11} \times 210 \times 4^{11} \times 7^{11} = 3.57 \times 10^{21}$  Solutions.

#### 2.2.2. Fitness Function

A fitness function is required to evaluate the quality of each chromosome. Generally, the fitness function is derived from the objective function and used in successive processes (Garg, 2010). The objective function of the proposed model is to minimise the total cost of the final product, which includes the product cost and the risk cost resulting from this reduction. To convert the objective function (minimum) illustrated in Equation 3 and 4 to a fitness objective (maximum), the fitness function is calculated as follows Equation 5:

$$F = \frac{100}{(C_T + 1)} = \frac{100}{\left(\left(\sum_{i=1}^{N_P} C_{P_i} + \sum_{i=1}^{N_P} C_{R_i}\right) + 1\right)} \quad (5)$$

The reason of adding the constant 1 to the equation is to prevent dividing the value by zero.

### 2.3. Reproduction

In this stage, genetic operators such as crossover and mutation are used to explore the search space, in order to create new chromosomes (offsprings). The number of new chromosomes ( $N$ ) in the offspring pool is determined by the crossover rate ( $P_C$ ) and mutation rate ( $P_m$ ) parameters considering  $P_C + P_m = \text{constant}$ .

**Table 1.** Chromosome construction

X1										X2				X3				X4															
1	0	1	0	1	0	1	0	1	0	1	70	1	3	4	2	1	3	2	4	1	3	4	1	4	2	6	1	3	7	2	5	3	4

## 2.4. Crossover

The crossover operator, which is the basic operator of GA, mates two chromosomes to produce new chromosome (offspring). The key purpose of using crossover operator is to recombine the features of two parents selected randomly from the mating pool to produce a new chromosome (Abdelmaguid and Dessouky, 2006; Jalilzadeh *et al.*, 2009). During the crossover, one or more genes will be exchanged between this pair of chromosomes. The crossover points are usually determined in a random manner. Crossover rate is another significant parameter in GA approach. A higher, but not too higher crossover rate permits more exploration of the solution space and minimise the opportunities of setting at a local optima (Mohd-Lair, 2008). This study uses three crossover methods: One-cut point crossover, flat-crossover and extended-intermediate-crossover. For each generation, one method is randomly selected to produce varied offspring.

### 2.4.1. Mutation

Mutation is a background operator which creates spontaneous random changes in numerous chromosomes. It is used to enhance the algorithm exploratory character and to avoid the GA from being trapped in local optima. Mutation operator helps the GA to avoid premature convergence and find the optimal solution in the search space based on the mutation rate ( $P_m$ ) (Abdelmaguid and Dessouky, 2006; Leuveano *et al.*, 2012). The main purpose of mutation operator is to produce little perturbations on chromosomes to maintain the diversity in the population.

The mutation rate also has a significant role in GA development. Therefore, defining the most appropriate crossover rate is necessary. To ensure the feasibility of mutated chromosomes, three methods of mutation are used: Simple random mutation, random exchange mutation and random insertion mutation. In each generation, one mutation is randomly selected.

### 2.4.2. Selection

The selection operator implements the idea of “the survival of the fittest”. It is the process of selecting chromosomes from the solution pool into a mating pool for generating offspring (Abdelmaguid and Dessouky, 2006). Selection operator describes the process of choosing a part of chromosomes from the initial population to keep their features in the next generation (Gen and Cheng,

1997). The main objective of selection strategy is to choose only the best chromosomes from the parents and offspring pools that can represent the solution space (Mohd-Lair, 2008). In this study, four selection methods (Roulette wheel, Binary tournament, Elitist and Replacement selection methods) are tested and the results illustrate that the replacement selection method is the best method that can be used for the proposed GA. The replacement selection method proves that the best chromosome always passes to the next generation. To avoid premature convergence (maintain population diversity), the mechanism strategy is also applied to adjust crossover and mutation rates.

### 2.4.3. Simplified Example

The proposed mathematical model has been tested with a simple assembly process of an electric motor with hollow shaft. It uses multiple, identical operations to assemble 25 individual parts into the finished product. The proposed GA is used to find the optimal or near-optimal solution of the problem. It is assumed that a production system purchases raw materials in a fixed lot size from eleven different regular external suppliers. It is also assumed that in the case of one or more of external suppliers suffering disruption, the production system can procure its raw materials from seven local backup suppliers. The details of the simplified example are presented in (El Dabee *et al.*, 2013b). Based on some experiments using different values of the GA basic parameters, the appropriate input parameters of the proposed GA are as shown in **Table 2**.

New chromosomes are produced by using crossover and mutation. Therefore, the crossover produces new chromosomes equal  $N \times P_c$  offspring and the mutation produces  $N \times P_m$  offspring for each generation. By using the crossover rate of 0.8, the number of chromosomes is  $400 \times 0.8 = 320$  offspring and the number of new chromosomes produced using the mutation rate of 0.2 is  $400 \times 0.2 = 80$  offspring in each generation.

**Table 2.** Input GA parameters

Parameter	Rate
Crossover rate ( $P_c$ )	0.8
Mutation rate ( $P_m$ )	0.2
Population size ( $N$ )	400.0
Maximum generations ( $G_{max}$ )	5000.0
Generation stop ( $G_{max-Stop}$ )	30000.0

### 3. RESULTS

The proposed GA tests the four selection methods (Roulette wheel, Binary tournament, Elitist and Replacement selection methods). A comparison between the four selection methods is carried out to determine which method is more suitable than the others by using the GA parameters illustrated in **Table 2**. The RCGA is run 10 times for each method to obtain the optimum solution for minimising the total cost of the final product and the risk effects in JIT systems in a short time. **Table 3** illustrates the average values of the best iterations, computation time, customer demand, product cost, risk cost, total cost (maximum and average) and fitness function (maximum and average) for each selection method.

### 4. DISCUSSION

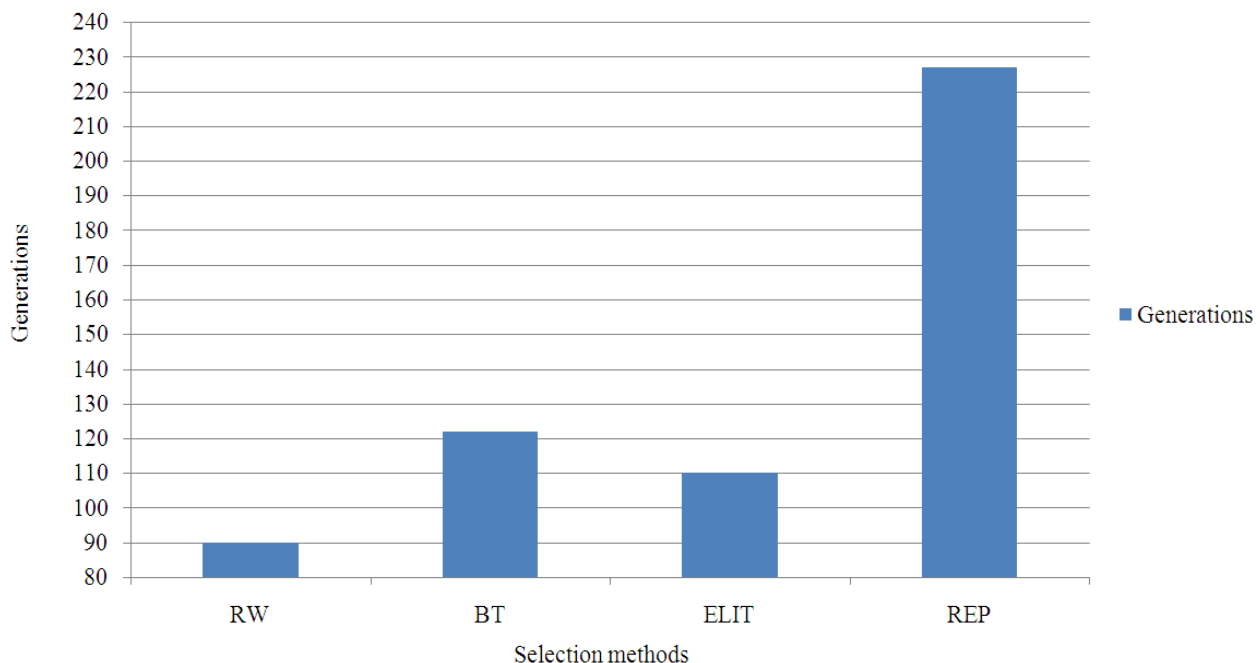
The computational results illustrated in **Table 3** indicates that the parameters  $d_p$ ,  $C_{pt}$ ,  $C_R$ ,  $C_{T-max}$  and  $F.F_{max}$  for all selection methods have the same values. However, the other values, which are the number of iterations (iter.), computation time ( $T$ ),  $C_{T-ave}$  and  $F.F_{ave}$  are different. These values show that the replacement selection method is the best methods compared with the other methods by achieving the higher average value of fitness values and minimum total cost.

**Figure 1** depicts the comparison of the average of maximum generations of the ten runs between the four selection methods. It is clear that using the replacement method, the average value of maximum generations is higher than the others.

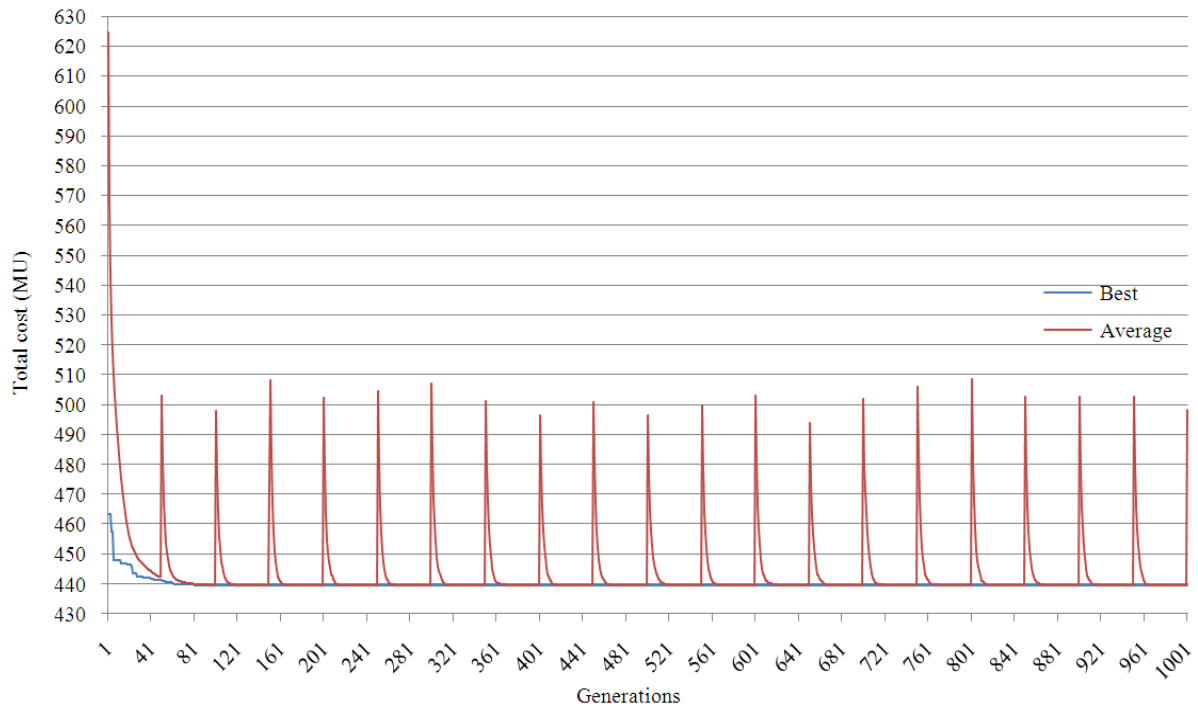
**Fig. 2** shows the average and best costs of all chromosomes related to the number of GA iterations. The injection strategy is used to maintain the population diversity. Form **Fig. 2**, it is clear that the injection strategy has caused a fluctuation on the average values since new chromosomes that may have lower fitness values are injected into the population.

Using the best iteration of the ten runs for each selection method, **Fig. 3** illustrates a comparison between the optimum solutions obtained by the four selection methods. It can be seen that the maximum generation is by using the replacement selection method.

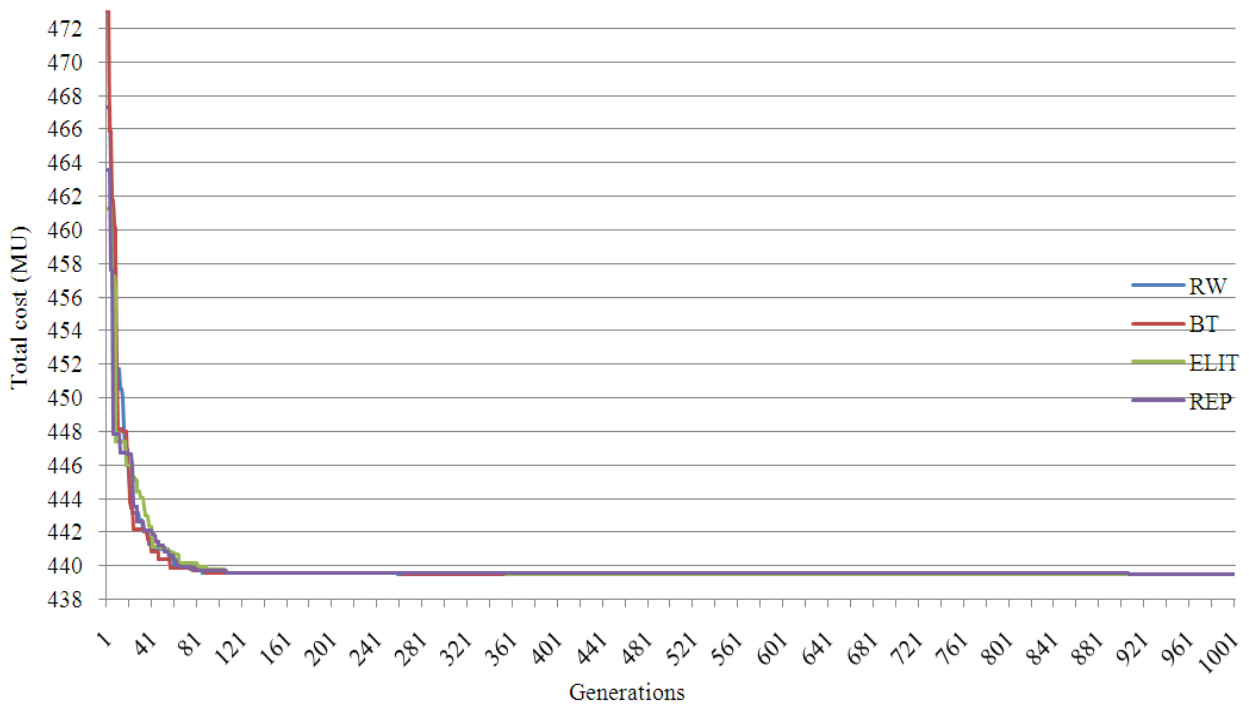
Adopting the appropriate selection method (replacement method), the GA was run 10 times and from the obtained results, the optimum solution is achieved after 360 generations and the best chromosome is shown in **Table 4**. The findings of the proposed GA are listed in **Table 5** including the number of iterations, the required time, the customer demand and the cost types.



**Fig. 1.** Comparison of the average generations between the four selection methods



**Fig. 2.** The average and best cost of the final product



**Fig. 3.** The best generation for each selection method





- El Dabee, F., R. Marian and Y. Amer, 2013c. A novel model for simultaneously minimising costs and risks in just-in-time systems using multi-backup suppliers: Part 1-modelling. *World Acad. Sci. Eng. Technol.*, 7: 148-155.
- Farahani, R. and M. Elahipanah, 2008. A genetic algorithm to optimise the total cost and service level for just-in-time distribution in a supply chain. *Int. J. Product. Econ.*, 111: 229-243. DOI: 10.1016/j.ijpe.2006.11.028
- Garg, P., 2010. Evolutionary computation algorithms for cryptanalysis: A study. *Int. J. Comput. Sci. Info. Sec.*, 7: 1-5.
- Ghasimi, S., R. Ramli and N. Saibani, 2014. A genetic algorithm for optimizing defective goods supply chain costs using JIT logistics and each-cycle lengths. *Applied Math. Modell.*, 38: 1534-1547. DOI: 10.1016/j.apm.2013.08.023
- Geem, Z., K. Lee and Y. Park, 2005. Application of harmony search to vehicle routing. *Am. J. Applied Sci.*, 2: 1552-1557. DOI: 10.3844/ajassp.2005.1552.1557
- Gen, M. and R. Cheng, 2000. *Genetic Algorithms and Engineering Optimization*. 1st Edn., John Wiley and Sons, New York, ISBN-10: 0471315311, pp: 495.
- Gen, M. and R. Cheng, 1997. *Genetic Algorithms and Engineering Design*. 1st Edn., John Wiley and Sons, New York, ISBN-10: 0471127418, pp: 411.
- Ghadge, A., S. Dani and R. Kalawsky, 2012. Supply chain risk management. *Int. J. logist. Manage.*, 23: 313-339. DOI: 10.1108/09574091211289200
- Go, T., D. Wahab, M. Rahman and R. Ramli, 2010. A design framework for end-of-life vehicles recovery: Optimization of disassembly sequence using genetic algorithms. *Am. J. Environ. Sci.*, 6: 350-356. DOI: 10.3844/ajessp.2010.350.356
- Goren, H., R. Tunali and S. Jans, 2010. A review of applications of genetic algorithms in lot sizing. *J. Intell. Manufact.*, 21: 575-590. DOI: 10.1007/s10845-008-0205-2
- Gupta, D. and S. Ghafir, 2012. An overview of methods maintaining diversity in genetic algorithms. *Int. J. Emerg. Technol. Adv. Eng.*, 2: 56-60.
- Hokoma, R., M. Khan and K. Hussain, 2010. The presentstatus of quality and manufacturing management techniques and philosophies within the Libyan iron and steel industry. *TQM J.*, 22: 209-221. DOI: 10.1108/17542731011024309
- Jalilzadeh, S., H. Shayeghi, M. Mahdavi and H., 2009. A ga based transmission network expansion planning considering voltage level, network losses and number of bundle lines. *Am. J. Applied Sci.*, 6: 987-994. DOI: 10.3844/ajas.2009.987.994
- Jianhua, W. and H. Xianfeng, 2010. A hybrid genetic algorithm for agile supply chain scheduling optimization. *Proceedings of the 2nd International Conference on Future Computer and Communication*, May 21-24, IEEE Xplore Press, Wuhan, 1: 396-400. DOI: 10.1109/ICFCC.2010.5497760
- Kannaiah, S., J. Thangavel and D. Kothari, 2011. A genetic algorithm based multi objective service restoration in distribution systems. *J. Comput. Sci.*, 7: 448-453. DOI: 10.3844/jcssp.2011.448.453
- Krajewski, L., L. Ritzman and M. Malhotra, 2013. *Operations Management: Processes and Supply Chains*. 10th Edn., Pearson, Upper Saddle River, ISBN-10: 0132807394, pp: 651.
- Kumar, T., S. Singh and C. Gupta, 2012. Genetic algorithm based multi product and multi agentinventory optimization in supply chain management. *Int. J. Model. Optimizat.*, 2: 653-657. DOI: 10.7763/ijmo.2012.v2.203
- Leuveano, A., F. Jafar and M. Muhamad, 2012. Development of genetic algorithm on multi-vendor integrated procurement-production system under shared transportation and just-in-time delivery system. *Proceedings of the 2nd International Conference on Uncertainty Reasoning and Knowledge Engineering*, Aug. 14-15, IEEE Xplore Press, Jalarta, pp: 78-81. DOI: 10.1109/URKE.2012.6319589
- Littidej, P. and S. Suttayamully, 2009. A minimizing cost of transport gabage in nonsung district nakhon ratchasima province. Thailand. *Am. J. Applied Sci.*, 6: 285-289. DOI: 10.3844/ajas.2009.285.289
- Mahmudy, W., R. Marian and L. Luong, 2012. Solving part type selection and loading problem in flexible manufacturing system using real coded genetic algorithms: Part I: Modelling. *World Acad. Sci. Eng. Technol.*, 6: 695-701.
- Malhotra, R., N. Singh and Y. Singh, 2011. Genetic algorithms: Concepts, design for optimization of process controllers. *Comput. Inform. Sci.*, 4: 39-54. DOI: 10.5539/cis.v4n2p39
- Mansouri, S., 2005. A multi-objective genetic algorithm for mixed-model sequencing on JIT assembly lines. *Eur. J. Operat. Res.*, 167: 696-716. DOI: 10.1016/j.ejor.2004.07.016
- Marian, R., 2003. *Optimisation of assembly sequences using genetic algorithms*. PhD. Thesis, University of South Australia.
- Mohd-Lair, N.A., 2008. *An integrated model foroptimising manufacturing and distribution networkscheduling*. PhD Thesis, University of South Australia.
- Murphy, P. and D. Wood, 2011. *Contemporary Logistics*. 10th Edn., Prentice Hall, Upper Saddle River, ISBN-10: 0132479036, pp: 329.

- Paksoy, T. and C. Chang, 2010. Revised multi-choice goal programming for multi-period, multi-stage inventory controlled supply chain model with popup stores in guerrilla marketing. *Applied Math. Modell.*, 34: 3586-3598. DOI: 10.1016/j.apm.2010.03.008
- Park, Y.J., S. Chun and B. Kim, 2011. Cost-sensitive case-based reasoning using a genetic algorithm: Application to medical diagnosis. *Artificial Intell. Medic.*, 51: 133-145. DOI: 10.1016/j.artmed.2010.12.001
- Radhakrishnan, P., V. Prasad and M. Gopalan, 2009. Optimizing inventory using genetic algorithm for efficient supply chain management, *J. Comput. Sci.*, 5: 233-241. DOI: 10.3844/jcssp.2009.233.241
- Rawal, R. and K. Inamdar, 2014. Review on various optimisation techniques used for process parameters of resistance spot welding. *Int. J. Current Eng. Technol.*, 3: 160-164.
- Scavino, E., D. Wahab, H. Basri, M. Mustafa and A. Hussain, 2009. A genetic algorithm for the segmentation of known touching objects. *J. Comput. Sci.*, 5: 711-716. DOI: 10.3844/jcssp.2009.711.716
- Sivanandam, S. and S. Deepa, 2008. *Introduction to Genetic Algorithms*. Springer Berlin Heidelberg, New York, ISBN-10: 978-3-540-73189-4, pp: 165-209.
- Tasan, S. and S. Tunali, 2008. A review of the current applications of genetic algorithms in assembly line balancing. *J. Intell. Manufact.*, 19: 49-69. DOI: 10.1007/s10845-007-0045-5
- Torabi, S., S.F. Ghomi and B. Karimi, 2006. A hybrid genetic algorithm for the finite horizon economic lot and delivery scheduling in supply chains. *Eur. J. Operat. Res.*, 173: 173-189. DOI: 10.1016/j.ejor.2004.11.012
- Wang, K. and Y. Wang, 2008. Applying genetic algorithms to optimize the cost of multiple sourcing supply chain systems-an industry case study. Springer Berlin Heidelberg, 355-372. DOI: 10.1007/978-3-540-76286-7\_16
- Wong, W., C. Kwong, P. Mok and W. Ip, 2006. Genetic optimization of JIT operation schedules for fabric-cutting process in apparel manufacture. *J. Intell. Manufact.*, 17: 341-354. DOI: 10.1007/s10845-005-0007-8
- Yussof, S., R. Razali and O. See, 2011. An investigation of using parallel genetic algorithm for solving the shortest path routing problem. *J. Comput. Sci.*, 7: 206-215. DOI: 10.3844/jcssp.2011.206.215
- Zhu, G. and S. Kwong, 2010. Gbest-guided artificial bee colony algorithm for numerical function optimization. *Applied Math. Comput.*, 217: 3166-3173. DOI: 10.1016/j.amc.2010.08.049