

# Cross-Comparison of Evolutionary Algorithms for Optimizing Design of Sustainable Supply Chain Network under Disruption Risks

Atiya Al-Zuheri<sup>1</sup>

<sup>1</sup> University of Technology, Baghdad, 10066 Iraq  
e-mail: [atiya.a.jiryo@uotechnology.edu.iq](mailto:atiya.a.jiryo@uotechnology.edu.iq)

## ABSTRACT

Optimization of a sustainable supply chain network design (SSCND) is a complex decision-making process which can be done by the optimal determination of a set of decisions and constraints such as the selection of suppliers, transportation-related facilities and distribution centres. Different optimization techniques have been applied to handle various SSCND problems. Meta- heuristic algorithms are developed from these techniques that are commonly used to solving supply chain related problems. Among them, Genetic algorithms (GA) and particle swarm optimization (PSO) are implemented as optimization solvers to obtain supply network design decisions. This paper aims to compare the performance of these two evolutionary algorithms in optimizing such problems by minimizing the total cost that the system faces to potential disruption risks. The mechanism and implementation of these two evolutionary algorithms is presented in this paper. Also, using an optimization considers ordering, purchasing, inventory, transportation, and carbon tax cost, a numerical real-life case study is presented to demonstrate the validity of the effectiveness of these algorithms. A comparative study for the algorithms performance has been carried out based on the quality of the obtained solution and the results indicate that the GA performs better than PSO in finding lower-cost solution to the addressed SSCND problem. Despite a lot of research literature being done regarding these two algorithms in solving problems of SCND, few studies have compared the optimization performance between GA and PSO, especially the design of sustainable systems under risk disruptions.

**Keywords:** comparison, disruption risk, genetic algorithm, particle swarm optimization, sustainable supply chain design.

## INTRODUCTION

Currently, the supply chain outlines the steps necessary to deliver a product or service to the end user and it includes suppliers, manufacturers and retailers who distribute the product to the end customer [1]. Supply chains include every business that comes into contact with a specific product, including companies that collect and deliver parts to the manufacturer. It is a critical process because an optimized supply chain leads to lower costs and a quick production cycle [2]. The design of a supply chain network is a strategic decision that plays an important role in the performance of the supply chain and its competitive advantages. Over time, the network's design decision is also affected in other decisions regarding operational and tactical level of the supply

chain [3]. Proper selection of suppliers, facilities and equipment, distribution centres are considered among the most prominent major decisions in the design of the supply chain network. These decisions directly or indirectly impact upon other decisions within the lower levels of the organization. Consequently, any changes in them may cause increased costs in the entire network [4].

The sustainable supply chain network (SSCN) refers to integrating green environmental processes into the traditional supply chain. This could include processes such as product design, material selection, manufacturing, production, operation and end-of-life management. While the specific goal of SSCM is often to reduce CO<sub>2</sub> emissions, reducing production costs has many tangible benefits for the organization. As a result of the growing importance of sustainability, it was considered one of

the main objectives in designing the supply chain network [5-8]. As supply chain system is a multi-objective system, most of these objectives tend to be in conflict [9]. Such examples of these conflict objectives are: maximum sustainability in the operating system, maximum utilization of resources, minimum lead time and minimum cost.

Abo-Hamad and Arisha [10] reviewed the literature related to applications of optimization approaches in the supply chain. GA has been proposed to derive solutions through the optimization process for different types of SCM problems. Jauhar and Pant [11] analyzed around 220 papers that applied GA for this purpose. On the other hand, particle swarm optimization algorithms have been successfully used in various areas of SCND. A brief review of this research work is given in [12]. In addition, combining this algorithm with other algorithms to produce hybrid algorithms has been introduced in many research papers; Kuo and Han [13], and Soleimani et al. [14].

Despite a lot of research literature being done regarding these two algorithms in solving problems of SCND, few studies have compared the optimization performance between GA and PSO, especially the design of sustainable systems under risk disruptions. As such, the aim of this paper is to compare computational performance of both PSO and GA algorithms in optimizing SCND by minimizing the total cost including a carbon tax under some given constraints. The performance comparison is based on the convergence rate of the algorithm.

The remainder of this paper is structured as follows. In section 2, a statement of the problem and the background about the modelling process to this problem is given. Section 3 presents a brief description of the GA and PSO algorithms and the comparison between them. Details of the case study used to perform the comparison are presented in section 4. Section 5 includes reporting the results of applying the algorithms on SSCND as well as a discussion of these results. Finally, in section 6, we draw some conclusions and recommendation for future research.

## STATEMENT OF THE PROBLEM

Based on Figure 1 the supply chain network studied in this paper distributes goods among customers from distribution centers. The typical manufacturer of a green supply chain encourages suppliers to adopt green practices, environmental management systems, etc. Then, products, after being defective, are returned to the supply chain and after examination the recoverable products are sent to recovery centers and the remains are sent to material customers. In recovery centers, after maintaining the products, they are returned to distribution centers to be sent to customer zones. In this paper we consider a problem of optimizing design of the supply chain network involving three main levels including the supplier, the manufacturer, and the customer. In order to take into consideration the difference in the price of the same product in the global marketing, it is assumed here that the distribution network consists of supplying raw materials from external suppliers. This assumption is made based on the basis that ensuring product quality, availability and affordability for the customers requires building adaptable and flexible supply chain networks. Thus, the materials that are ordered come from numerous suppliers to the assembly systems where the final product (the electric motor in our case study) will be assembled and then be ready for sale wholesale or retail outlets. Further, and to meet the requirements of adopting JIT strategy, the raw materials entering the assembly system are instantaneously replenished. At the time the products have been assembled they will be sent immediately to the final customer. In doing so, the costs related to the storage of raw materials as well as the storage of final products will not be taken into account. Wholesalers or retailers may consider from the prospected customers. Disruption risks of supply chain operations that arising from occurring economic crises, poor weather, natural or man-made disasters or a combination of any other risks, can affect the flows of raw materials

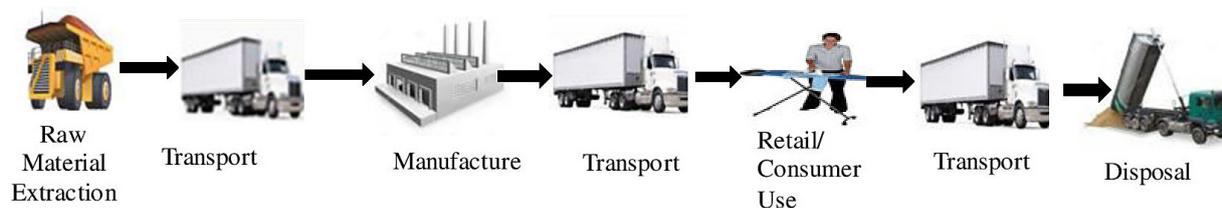


Fig. 1. The structure of the typical supply chain management system

or parts from external suppliers. Most managers know that all of these unforeseen disruptions can wreak havoc on logistic operations, and the goal always is reducing the impact of their systems by these unexpected disruptions.

To cope with disruption risks, companies in their risk management plans usually organize their production systems to procure their raw materials required to produce the finished products from local backup supplier. However, such a plan may be costly due to higher price links with a low occurrence risk and with shorter lead time [15]. An optimization cost model is adopted in this paper for simultaneously reducing the cost and risk effects in organizations that use a JIT strategy. In addition, the model considers the rate of CO<sub>2</sub> emission resulting from transportation in the supply chain network. Also, in order to satisfy customer requirements, the presented optimization model takes into account scenarios including orders from a set amount of input materials to be shipped by both external and local suppliers using different transportation modes; waterways, railways, roadways and airways [16]. There are a number of key criteria that companies take into consideration when selecting suitable mode of transport to integrate the raw materials into the production system. Among these, the key criteria that have been considered in this paper are the lead time, the transportation costs, capacity, and CO<sub>2</sub> emissions.

## SOLUTION PROCEDURES AND ALGORITHMS

A solution to the addressed problem is to create a sustainable supply chain network design (SSCND) under disruption risks that is based on the optimization model presented in study of Al-Zuheri and Vlachos [17] which is an extended version of the model presented by El Dabee et al. [15]. The optimal network design can be illustrated with the help of total costs, involving decision variables: the number of external suppliers that are employed to supply raw materials to the production system, the customer demand for the final product per day, critical transportation measurement of raw materials shipped using transportation mode and the quantity of raw material  $i$  ordered in each patch required to produce the final product per week.

The next subsections briefly describe the structure and mechanism of the proposed optimization model and also evolutionary algorithms

(GA and PSO) which is used in a comparison performance to optimize the design sustainable supply chain network under disruption risks.

### The optimization model

The model used to solve this research problem which will be used in both algorithms (GA & PSO), the parameters, indexes and decision variables are detailed in the studies of El Dabee et al. [15] and Al-Zuheri and Vlachos [17]. The first objective of that model was to minimize the total product cost and the risk effect in the supply chain system which adopts JIT strategy “ $C_T$ ”. It was also set for minimizing the cost of carbon tax “ $C_{tax}$ ”. So, the ultimate objective has been formulated as follows:

$$\text{Minimise } C_{Total} = \sum (C_T + C_{tax}) \quad (1)$$

The optimization model included calculation of different costs. One of these costs is the cost of purchasing materials from various suppliers “ $C_{RM}$ ”, the laborers’ cost “ $C_L$ ”, which represents the wages paid to workers for performing their duties in any company in a time unit. The cost of utilities “ $C_U$ ” is taken into consideration as well. The total cost is the sum of six main types of costs: Ordering Cost “ $C_O$ ”, Holding Cost “ $C_H$ ”, Purchasing Cost “ $C_P$ ”, Transportation Cost “ $C_{tr}$ ”, Duties Cost “ $C_D$ ”, Transfer Price Cost “ $TP$ ”. As mentioned above, details about those costs in terms of calculations, syntax and semantic, can be found in research conducted by El Dabee et al. [15] and Al-Zuheri and Vlachos [17].

### Genetic algorithms

Genetic algorithms mimic the mechanism of genetics and natural selection (survival of the fittest) and natural genetics described by Charles Darwin [18]. Holland (1975) introduced GA in the 1960’s and 1970’s as a technique to evolve an optimal solution from a population of initial feasible solutions available for solving an optimization problem. Genetic algorithms are useful and effective when the research space is large or complex or when it is difficult to use traditional research methods to solve the problem and this algorithm has proven very successful in many fields and its use has spread. To produce an optimal generation and by repeating the genetic cycle, the quality of the offspring will gradually improve. The basic stages of the algorithm are shown in Figure 2.

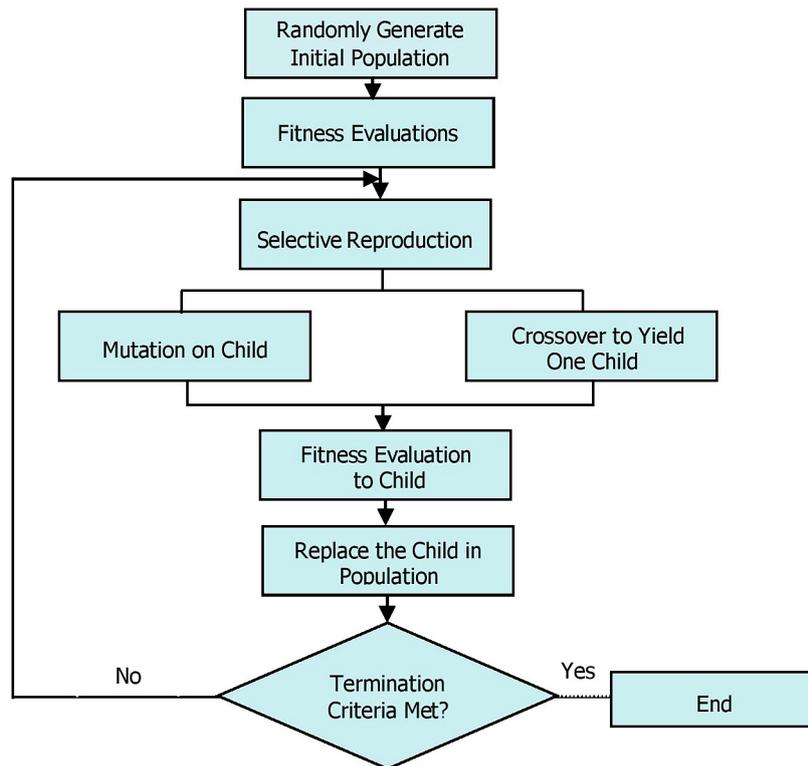


Fig. 2. Flow chart of GA

**Particle swarm optimization**

This algorithm was developed by Eberhart et al. [19]. It was inspired by the social behavior of flocks of birds or fish groups as they move from one place to another. It simulates the natural behavior of population swarms. Actually, within the PSO algorithm the optimal solution is searched for by tracing and following the current best-looking particles - similar to bee and ant behaviors for example.

Actually, within the PSO algorithm the optimal solution is searched for by tracing and following the current best-looking particles - similar to bee and ant behaviors for example. There are many similarities between PSO and other evolutionary computation techniques, for example “GA”. However, unlike the GA algorithm, the PSO does not have evolutionary tools such as crossover and mutation. The particles fly in the search space where the velocity modulation of these particles is bound to its own velocity which is continuously dependent on the behavior of the swarm so that the particles have a tendency. The best location can be taken by the particle according to the following steps:

*Step 1 Initialization*

The setting of the velocity of all particles, as well as their location, is random and within pre-determined rates.

*Step 2 Updating of particle velocity*

All particle velocities are updated at each iteration. The updating process is conducted according to the following:

$$\vec{V}_i = w\vec{V}_i + c_1R_1(\vec{l}_{i,best} - \vec{l}_i) + c_2R_2(\vec{p}_{i,best} - \vec{l}_i) \tag{2}$$

Where  $\vec{V}_i$  and  $\vec{l}_i$  are the velocity and location of particle  $i$ , respectively. The  $\vec{l}_{i,best}$  and  $\vec{p}_{i,best}$  represent the location with the “best” objective value which is found by particle  $i$  and the entire population respectively. Also,  $w$  is the parameter which controls the “flying dynamic” of the particle, when we consider it as a bird.  $C1$  and  $C2$  are factors used to control the related weighting of corresponding terms. These factors compromise the inevitable trade-off between exploitation and exploration.  $R1$  and  $R2$  are continuous random variables defined in a range  $[0, 1]$ . Having these two random variables in the PSO structure will enable stochastic searching. After the update and to avoid violent random walking, the  $\vec{V}_i$  must be checked and secured within a predetermined range.

*Step 3 Updating of location*

With the assumption that there is a unit time interval between successive iterations, the updating of all particle locations is based on the following:

$$\vec{l}_1 = \vec{l}_i + \vec{V}_i \quad (3)$$

When the update process has finished,  $\vec{l}_i$  should be checked and limited within the allowed range.

**Step 4 Updating of memory**

It has been identified as updating  $\vec{l}_{i,best}$  and  $\vec{p}_{i,best}$  if the condition is met.

$$\vec{l}_{i,Best} = \vec{l}_i \quad \text{if } f(\vec{l}_i) > f(\vec{l}_{i,Best}) \quad (4)$$

$$\vec{p}_{i,Best} = \vec{p}_i \quad \text{if } f(\vec{p}_i) > f(\vec{p}_{i,Best}) \quad (5)$$

Here,  $f(\vec{x})$  is the objective function that has to be maximized.

**Step 5 Terminating criteria**

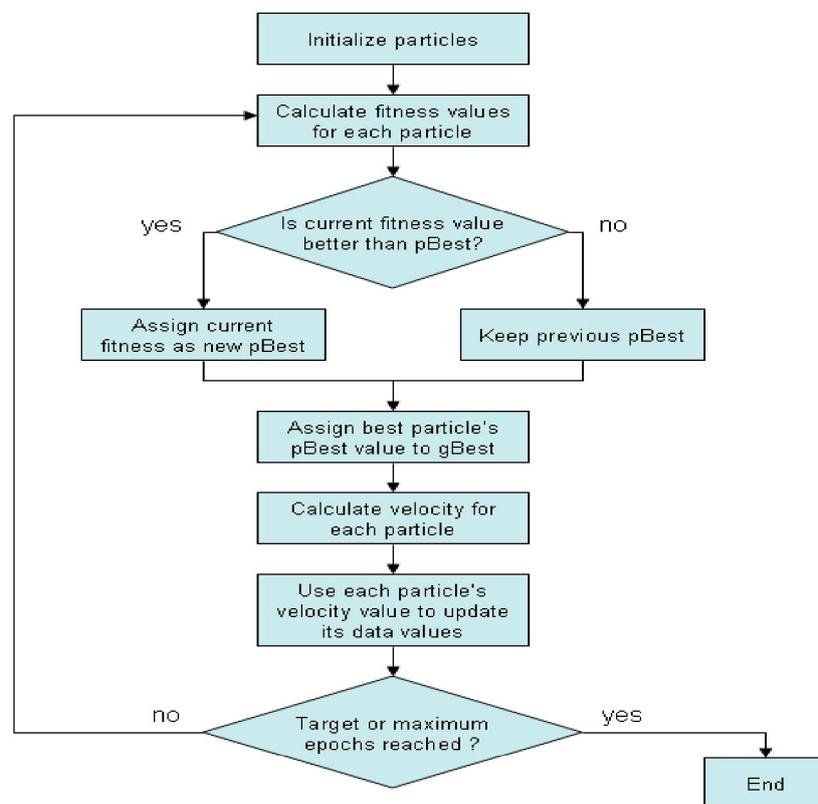
The algorithm often uses repetition to execute steps (2 to 4) until certain termination conditions are met. Termination conditions for PSO can be set using one the following criteria: (i) the number of iterations reaches a predefined number, and (ii) the progress remains unchanged for a certain number of iterations. Once terminated, the algorithm then reports that the optimal solution has values of  $\vec{p}_{i,Best}$  and  $f(\vec{p}_{i,Best})$ . Figure 3 illustrates the overall steps

of PSO. The comparison of the PSO with the GA will be explained in the forthcoming section.

**Comparison of the PSO Algorithm with the genetic algorithms**

Most of the development algorithms have two main steps. The first step is the random generation of the initial population which is required to run for subsequent generations. The second step is the calculation of the fitness value for each solution which depends directly on the “distance” from the optimum value. When the requirements are met the stoppage of the algorithms’ run occurs, otherwise the run will return to the step 2. Based on these steps, it was found that the PSO algorithm shares several common points with the GA algorithm. Both algorithms start generating random clusters. Both have also reliable dependency values for the aggregate evaluation. In addition, the populations are updated in the two algorithms as well as searching for the optimum value using random techniques. However as other techniques optimize none of them can guarantee a definite progress in achieving the optimal solution.

The PSO algorithm does not possess the same parameters as genetic algorithms such as



**Fig. 3.** The flowchart of the conventional PSO Algorithm (Eberhart and Kennedy, 1995)

crossover and mutation. The particles within the PSO algorithm update themselves by taking advantage of the internal speed value. The elements also have memory which is an important component of the algorithm. Within the GAs genetic algorithm, the chromosomes share their information with each other. Thus, the population moves in one population towards the optimal area. Within the PSO algorithm only each best group gives the information that is important to the rest. Thus, these algorithms have a (one-way) mechanism for sharing information. Accordingly, the evaluation process depends mainly on its search for better solutions. All elements within the PSO algorithm tend to converge rapidly toward these improved solutions, even within small (for the most part) local clusters. At this point, we have completed this brief comparison of our algorithm with genetic algorithms. A demonstration will follow in the next sections using a real case example.

### CASE EXAMPLE

As we already stated, the optimization model proposed in this paper applied to the decision – making of simple assembly process for hollow shaft electric motor which was presented in the research of El Dabee et al. [15]. The assembly process includes multiple and identical operations to assemble twenty-five individual parts into the finished product ( $N_p = 25$ ). The main components of this motor are illustrated in Figure 4.

It is assumed that a production system purchases raw materials in a fixed lot size from eleven different regular external suppliers ( $N_{SE} = 11$ ). In such a JIT system, when needed, raw materials are delivered at fixed time intervals and they are supplied by those external suppliers in a limited time. The production system intended for the assembly of the electric motor includes five processes. In order to launch these operations, the company has used five workers, each one of them, managing and running one of these operations. The daily shift hours  $N_h$  is eight hours per day, five days per week. The hourly wage per worker  $C_{w_i}$  was fixed at 14 monetary units (MU/hour). The utilities cost  $C_U$  is assumed to be equal to 10% of raw material cost. It is also assumed in this case that one or more external suppliers suffer disruption adversely affecting the production system. Seven local backup suppliers ( $N_{SLB} = 7$ ) could provide raw materials with a lower risk factor and a shorter lead time but with a higher cost. The production facility produces 70 units per day. Raw materials, in the first instance, are procured from regular external suppliers  $S_E$ . If the flow of goods is disrupted from one or more of its regular suppliers, the company resorts to local suppliers ( $S_{LB}$ ) to ensure the continuous flow of raw materials. The suppliers lead time directly affects the order, so it is imperative to take this into consideration when meeting customers’ needs during fixed intervals and at normal times. Any time delay in receiving raw materials to the production system can initiate many risk factors such as economical,

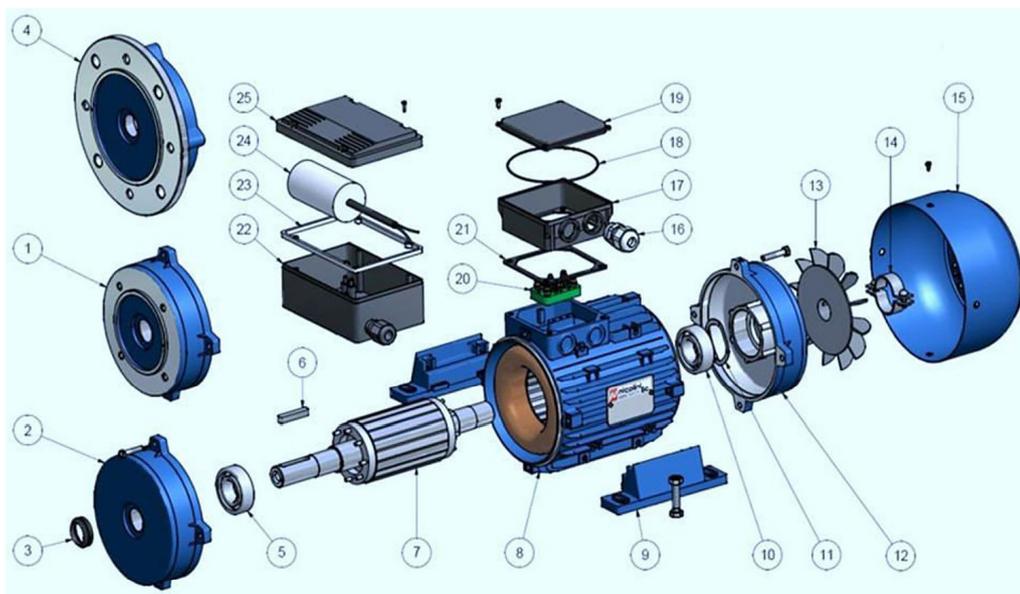


Figure 4. Motor - electric with hollow shaft

legal, operational, social, political and physical, and these will directly affect and disrupt the production system and all its supply chain processes. The end-customer purchases the final product at 485 MU. The next step is to identify the supply chain risks facing the production facility. As we previously stated, more on the case example details can be found, in El Dabee et al. [15].

## RESULTS AND ANALYSIS

### Tuning GA parameters using DOE approach

GA performance is largely dependent on its basic parameters and these are important factors in controlling the power of GA. In order to determine the “best” GA parameter combinations to be used in the presented GA, a design of experiments (DOE) approach has been conducted. Full factorial design has been used with a two level four factor design for the tuning of GA parameters such as crossover rate ( $P_c$ ), mutation rate ( $P_m$ ), population size ( $N$ ) and the number of generations ( $G_{max}$ ). Table 1 shows the upper and lower levels of GA parameters. The GA termination criterion occurs at a maximum generation number which is illustrated in Table 1.

**Table 1.** Upper and lower levels of GA parameters

Level	Crossover rate $P_c$	Mutation rate $P_m$	Population size $N$	Number of generations $G_{max}$
Lower	0.15	0.15	100	1000
Upper	0.7	0.7	300	5000

**Table 2.** Computational results of the average values of 10 runs of the GA parameters

Exp. No.	$P_c$	$P_m$	$N$	$G_{max}$	$FF$	$FF$	$CT$	$CT(Ave)$	Average of C.G
1	0.15	0.70	100	1000	0.237	0.236	440.503	441.684	258.800
2	0.15	0.70	100	5000	0.237	0.233	440.503	449.159	336.300
3	0.15	0.70	300	1000	0.237	0.236	440.503	441.987	168.600
4	0.15	0.70	300	5000	0.237	0.230	440.503	453.730	210.000
5	0.15	0.15	100	1000	0.237	0.232	440.597	450.811	499.900
6	0.15	0.15	100	5000	0.237	0.232	440.503	450.731	396.400
7	0.15	0.15	300	1000	0.237	0.235	440.503	444.379	270.800
8	0.15	0.15	300	5000	0.237	0.232	440.503	449.784	264.700
9	0.70	0.70	100	1000	0.237	0.237	440.503	440.595	179.400
10	0.70	0.70	100	5000	0.237	0.237	440.503	440.665	430.900
11	0.70	0.70	300	1000	0.237	0.231	440.503	452.748	199.200
12	0.70	0.70	300	5000	0.237	0.237	440.503	440.702	196.600
13	0.70	0.15	100	1000	0.237	0.235	440.503	443.915	187.200
14	0.70	0.15	100	5000	0.237	0.237	440.503	441.193	632.200
15	0.70	0.15	300	1000	0.237	0.236	440.503	442.587	185.100
16	0.70	0.15	300	5000	0.237	0.237	440.503	440.846	171.000

### DOE results

Using a DOE approach and with considerable diversity in the parameters of GA which is shown in Table 2, sixteen experiments were carried out. Each experiment was repeated for 10 runs to confirm the credibility of the results. Referring to the optimization model introduced by El Dabee et al. [15], the decision variables of the problem used in the in the experiments are as follows. The number of external suppliers used for supplying raw materials to the production system ( $N_{SE}$ ) is 11. The customer demand for the final product per day ( $d_p$ ) is 70 unit/day. The critical transportation measurement of raw materials shipped using transportation mode ( $tm$ ) is 1, and the quantity of raw material  $i$  ordered in each patch required to produce the final product per week ( $Q_M$ ) is 350 unit/week. The results are summarized in Table 2.

The results presented in Table 2 incorporate the best and average fitness function ( $FF$  and  $FF(Ave)$ ) and the minimum and average total cost ( $CT$  and  $CT(Ave)$ ). The results also incorporate average value of 10 runs of the convergence generations ( $C.G$ ).

Rapid convergence has been noted in experiments 9 and 16 and the best GA solution is reached after 179 and 171 generations. These

experiments generate “super” GA solutions at an early generation, and they are so superior to other GA solutions that they inhibit the full exploration of the search space. The minimum fitness function value of 0.237 reached in experiment 179 was the lowest minimum value in the 16 experiments. It is clear from the results shown in Figure 5 that the appropriate rates of GA basic parameters that can be used to obtain the optimal solution to the SSCND problem and algorithms performance comparison are:  $P_c = 0.7$ ,  $P_m = 0.15$ ,  $N = 300$ , and  $G_{max} = 5000$ . At these parameter rates, the convergence of fitness function starts after 171 generations.

**Optimization of SSCND and comparisons using evolutionary algorithms**

For optimal purposes the design of a sustainable supply chain network investigated in this research, and consequently its performance comparison and implementations of GA and PSO evolutionary algorithms are coded and simulated using Java programming software. The specified parameters related to the GA and PSO are shown in Table 3. The fitness function which is presented in equation (1) is used to reflect the goodness of each algorithm.

The performance of these two algorithms was tested 10 times for each one to obtain the optimum solution to the identification problem in this research. The algorithms performance is generated while considering the parameters settings of GA and PSO algorithms which are listed in Table 3. The results are provided in Table 4 which shows the characteristics of the best run optimized by the two algorithms. The results confirm one of the GA’s drawbacks: it has an expensive computational cost. The developed GA reached the optimum solution after 223 iterations, while PSO achieved this after 126 iterations. It should be noted however that the fitness values are equalized but the total cost value of the final product with associated risk in JIT systems is highest for PSO. The improvement in performance of GA with the given fitness function is clearly attributed to the result of population diversity caused by adopted selection strategy.

Herein, the operators; crossover or mutation generates a new offspring which can move in different directions in solutions space and consequently enable GA to avoid trapping in local optimal areas. As can be shown in Table 4, PSO reaches faster towards the optimal solution design (126 iterations while GA 223 iterations) but both algorithms take the same time (around 48 seconds)

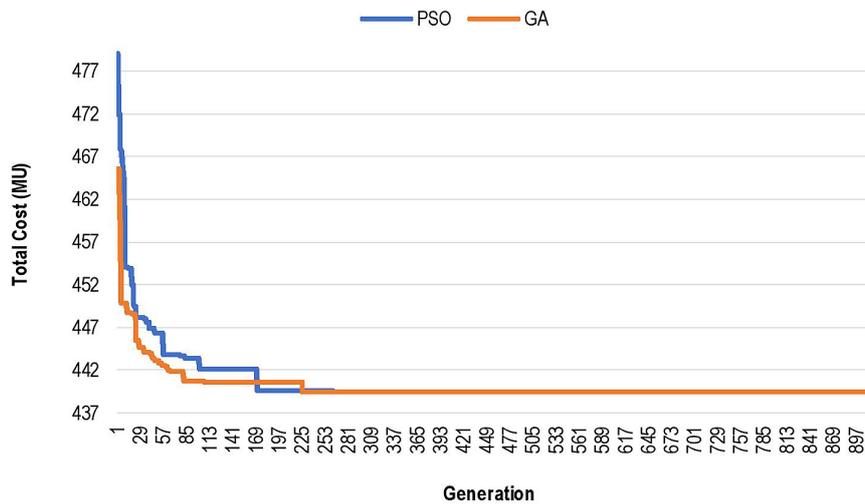
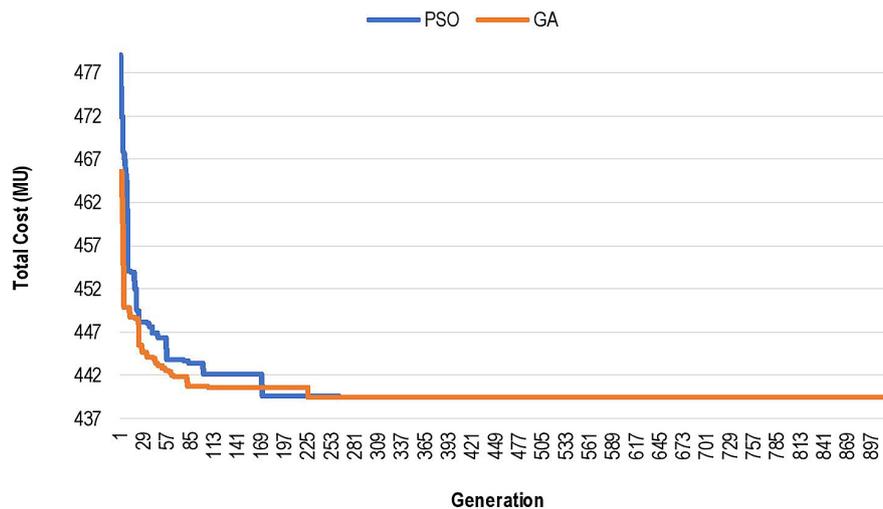


Figure 5. Minimum and average total cost of 16 experiments. Each experiment run for 10 times

Table 3. Parameters used in GA and PSO

GA Parameters	PSO Parameters
Crossover rate ( $P_c$ ): 0.7	$C_1, C_2$ cognitive and social acceleration coefficients: 2.0, 2.0
Mutation rate ( $P_m$ ):0.15	$W_{start}, W_{end}$ inertia weight coefficients: 0.9, 0.4
Population size ( $N$ ):300	swarm size ( $N$ ):300
Maximum generations ( $G_{max}$ ): 1000	Maximum generations ( $G_{max}$ ): 1000



**Figure 6.** Comparison of the steps in the improvement to minimize the total cost of best solution design to SSCN over GA and PSO generations

to determine their results. The PSO algorithm is appropriate to a system which is linear, time invariant and deterministic. However, supply chain networks are considered to be derived from stochastic systems - as detailed before - so none of these characteristics are perfectly achieved. The following out; Figure 6 is a typical GA and PSO simulation in search for an optimizing SSCND. Figure 6 shows that PSO could reach the optimal solutions faster than GA. However, GA was in a regular pattern of significant improvement during the search for optimal points. This consistency in GA performance is attributed to the three supportive operators: selection, crossover and mutation, while in the PSO the updating search is only depended on one parameter: velocity change. The detailed analysis of the two algorithms, as well as the effects of their different parameters cannot be explained in detail here due to the restrictions. The reader can find an extensive discussion of many of these points in the literature. In its effort to obtain the optimal solution, this algorithm relies on simulating the behavior of birds in the search for the best food. Therefore, randomness exists in any system which depends on this algorithm. This is because the system will consist of a random set of random solutions. Based on this beginning, the optimal solution is generated through the modernization of generations.

## CONCLUSIONS

Consider the current Covid-19 pandemic which has had a significant impact on the supply

chain network of many companies, as well as causing ongoing turmoil in the worldwide and adverse effects on the global supply chain and the possible disruptions to goods supplies. This paper was to answer the question which has raised regarding which evolutionary optimization algorithm can be used to design sustainable supply chains under such disruption risks. The authors compared the performances of two optimization techniques: genetic algorithms and particle swarm optimization to design a SSCN with the risk of disruptions. In this paper a numerical real-life case study was introduced to conduct the performance comparison of the algorithms. The comparison aimed at analyzing the performance to reach an optimal solution with lower cost. Both algorithms are based on population search methods and then the generated population is evolved from one generation to the next via a combination of deterministic or probabilistic rules. The results indicated that the GA is superior to the PSO in solving this kind of problem.

Recently, an optimal setting of GA parameters has been considered by an increasing number of research works although this setting in the PSO has not yet been considered. Future research can explore the possibility of integrating GA into PSO in order to enhance the performance of the algorithm.

## REFERENCES

1. Simchi-Levi D., Kaminsky P., Simchi-Levi E. Designing and managing the supply chain. Concepts, Strategies, and Case Studies. McGraw-Hill. Irwin. 2000.

2. Koc Ç. An evolutionary algorithm for supply chain network design with assembly line balancing. *Neural Computing and Applications*. 2019;28(11):3183–3195.
3. Gholian-Jouybari F., Paydar M.M., Hajiaghayi-Keshteli M., Fathollahi-Fard A.M. A bi-objective stochastic closed-loop supply chain network design problem considering downside risk. *Industrial Engineering & Management Systems*. 2017;16(3):342–362.
4. Devika K., Jafarian A., Nourbakhsh V. Designing a sustainable closed-loop supply chain network based on triple bottom line approach: A comparison of metaheuristics hybridization techniques. *European Journal of Operational Research*. 2014;235(3): 594–615.
5. Chaabane A., Ramudhin A., Paquet M. Design of sustainable supply chains under the emission trading scheme. *International Journal of Production Economics*. 2012;135(1):37–49.
6. Chen G., Govindan K., Golias M.M. Reducing truck emissions at container terminals in a low carbon economy: Proposal of a queueing-based bi-objective model for optimizing truck arrival pattern. *Transportation Research. Part E: Logistics and Transportation Review*. 2013;55:3–22.
7. Ardalan Z., Karimi S., Naderi B., Khamseh A. A. Supply chain networks design with multi-mode demand satisfaction policy. *Computers & Industrial Engineering*. 2016;90:108–117.
8. Hajiaghayi-Keshteli M., Fathollahi-Fard A.M. Sustainable closed-loop supply chain network design with discount supposition. *Neural Computing and Applications*. 2019;31(5): 5343–5377.
9. Min H., Zhou G. Supply chain modeling: Past, present and future. *Computers & Industrial Engineering*. 2002;43(1–2):231–249.
10. Abo-Hamad W., Arisha A. Optimisation methods in supply chain applications: A review. In: *Proc. of 12<sup>th</sup> Annual Irish Academy of Management Conference*, Galway Mayo Institute of Technology, Galway, Ireland 2009.
11. Jauhar S.K., Pant M. Genetic algorithms in supply chain management: A critical analysis of the literature. *Sādhanā*. 2016;41(9):993–1017.
12. Kadavevaramath R.S., Chen J.C.H., Shankar B.L., Krishnaswamy R. Application of particle swarm intelligence algorithms in supply chain network architecture optimization. *Expert Systems with Applications*. 2012; 39(11):10160–10176.
13. Kuo R.J., Han Y.S. A hybrid of genetic algorithm and particle swarm optimization for solving bi-level linear programming problem – A case study on supply chain model. *Applied Mathematical Modelling*. 2011;35(8):3905–3917.
14. Soleimani H., Govindan K., Saghafi H., Jafari H. Fuzzy multi-objective sustainable and green closed-loop supply chain network design, *Computers & Industrial Engineering*. 2017;109:191–203.
15. El Dabee F., Marian R., Amer Y. A novel optimization model for simultaneous cost-risk reduction in multi-suppliers just-in-time systems. *Journal of Computer Science*. 2013;9(12):1778–1792.
16. Murphy P.R., Wood D.F. *Contemporary Logistics*. Prentice Hall; 2004.
17. Al-Zuheri A., Vlachos I. A model for designing sustainable supply chain network under disruption risks and carbon tax charges. In: *Proc. of 12<sup>th</sup> ICLS2017 Conference*, Beijing, China, 2017.
18. Gen M., Cheng R. *Genetic Algorithms and Engineering Design*. John Wiley & Sons; 1997.
19. Eberhart R., Shi Y., Kennedy J. *Swarm Intelligence*, Morgan Kaufmann; 2001