International Journal of Innovation, Creativity and Change. www.ijicc.net
Volume 12, Issue 1, 2020

An Optimal Decision Making Process to Determine the Sequence of Products in Productive Sectors Using Genetic Algorithms

Ahmed Karim Jassim Jassim[a], Ali Hussein Hasan[b], aCollege of Administration and Economic, University of Thi-Qar, Iraq, bCollege of Computer science and IT, University of Sumer, Iraq, Email: [a]Ahmed.kareem@utq.edu.iq, [b]ali.husain@uos.edu.iq

The productive sectors at the present time are working in a competitive work environment, characterised by rapid development and change in the wants and needs of customers because of the opening of the country's markets to international and Arab companies, making the competition process of these companies very difficult in that the customers demand different goods and services to meet their needs. Also, these goods are high in quality and reliability, lowest in cost and also rapid in delivery and response. The production companies are working to develop their production lines so that they are more flexible to change and can produce multiple products on the same production line, with the possibility of development to cope with the changing needs of customers. The changes of the production line from the production of a product to another needs a setup time and this time may be greater than the production time, thus leading to delay in the delivery of demands, which affects the reputation of the company and leads to customer dissatisfaction. Therefore, setup time is considered to be an influential factor in the production process. The determining of product sequence that reduces setup time using traditional approaches needs a long time to take all the possibilities for the relay process, and then to choose the alternative that reduces setup time. This leads companies to use modern scientific and quantitative approaches of rational decision making, and not enough of the experience of the decision maker or the use of traditional approaches in the process of decision making. In this paper, we used Genetic algorithms (GA) in decision making, which is one of the modern quantitative approaches to find the global optimum solution. It has been used to determine the sequence of products and the least setup time for a given set of demands. The proposed approach was applied in the Ur Company/ wire winding factory. The results showed the efficiency of the proposed algorithm in decision making to determine the production sequence in this factory, where GA take into account the product which ended the previous demand, and which one will begin in the later demand. Therefore, will be making a strategic and comprehensive decision and not a local decision.

Key words: *Decisions making, products sequences, set-up time, genetic algorithm.*

Introduction

Thousands of business decisions are made every day. It is noted that not all the decisions will make or break the organisation; each one may add a measure of success or failure to the operations. Management scientists hold that education, scientific training and experience can improve a person's ability to make decisions. Scientific decision making rests upon organised principles of knowledge and depends largely upon the collection of empirical data and analysis of the data in a way that repeatable results will be obtained (Anil and Suresh, 2009).

Decision making for sequencing problems is the most impactful factor inside companies, where production line structure requires all products to pass through some workstations in the same sequence. The market required greater flexibility and a variety of products together, with the reduction of life cycles. These guides lead the companies to necessarily utilise from all available requirements, therefore decision making is required in interpreting this direction to produce multi-products in the same production line, thus fulfilling customer demands in the dynamic environment. The objective of decision makers in these companies is to reduce the setup time for all demands (Riddalls and Bennett, 2001). (Cheng et al., 2001) shows scheduling with batching is an NP-hard problem, and heuristics are better suited at solving a scheduling problem when batching is present, showing that single batch processor scheduling problems with multiple product families, setup times, and focussing on due date objectives, such as minimising lateness, is strongly NP-hard. (Pinedo, 2002) presented scheduling problems involving sequence-dependent setups is well known to be similar to that of the Travelling Salesperson Problem (TSP), which is known in the literature to be NP-hard. (Kreipl and Pinedo, 2004) provide insights into the use of planning and scheduling models in supply chain management, as well as into the information sharing and interactions that occur between the different types of models that are embedded in one system. Planning models have often been analysed in detail; scheduling models, on the other hand, have been studied less often within a supply chain management framework. This modelling is done in practice – there is a very strong emphasis on transportation costs and less of an emphasis on setup costs. (Zhu and Wilhelm, 2006) studied scheduling problems with sequence-dependent setups. Most research considers the reduction of setup costs and inventory holding costs. This study shows that any scheduling problem that has sequence-dependent setups, any performance objective (e.g., minimise the makespan, minimise total flow time, minimise lateness, etc.) is NP-hard. (Charnprasitphon, 2007) developed methodologies that are applied to two fundamental problems: the batch production scheduling problem for perishable products with sequence-independent setup times (BPP-SI) and sequence-dependent setup times (BPP-SD). The new models for both problems formulates them as a mixed-integer program (MIP) in discrete time.

(Jingxu, 2008) introduced that efficient production scheduling and sequencing are important to achieve the overall material supply, production, and distribution efficiency around the mixed-model assembly line in a supply chain, where production scheduling and finished goods distribution have been increasingly considered in an integrated manner to achieve an overall best efficiency.

(Clark et al., 2010) formulated sequencing and lot sizing with non-triangular setup times based on asymmetric travelling salesman problem (ATSP) at an animal feed plant. To solve the model, optimal solution methods based on iterative sub tour elimination and patching are developed. (Salmasi et al., 2011) developed a mathematical programming model in order to minimise the total flow time on the flow-shop group scheduling (FSGS) problem for solving large size issues. After having defined a wide benchmark of test cases arising from real world manufacturing environments, the authors completed an extensive comparison among the proposed metaheuristics, from which the outperforming results of the ant colony approach clearly emerged. (Costa et al., 2013) analysed a flow shop sequence-dependent group scheduling problem with limited inter-operational buffer capacity, truly observed in the inspection department of a company producing electronic devices. The authors proposed a matrix-encoding (GA). (Patricia, 2014) applied the travelling salesperson problem to many other fields such as logistics, planning and manufacturing, where the ultimate goal is to find the optimal path given a set of distances. This study presents different approaches to solving scheduling problems when batching and with sequence-dependent setups. These solution approaches range from exact methodologies to heuristic methodologies. These various solution approaches can be used in order to improve performance metrics such as a number of late jobs, maximum lateness, and deviation from job due dates. (Laith et al., 2015) utilised the modified assignment method based on the goal programming method to determine the optimal product sequence-dependent on setup cost and/or setup time in a single demand, that consists of multi-products. This method can find the optimum solution directly and also help the change in priority of the goals of the company, leading to multiple alternatives. (Mohammadi and Mohammadi, 2016) studied the flow-line of the manufacturing cell with sequence-dependent parts family setup times to minimise the makespan criteria. In this paper we use the genetic algorithm to determine an optimal sequence of all products in which the demands are implemented in the production department after determining the sequence of these demands by the planning department: then a decision can be made. The algorithm is tested and applied in the wire factory of Ur company.

Problem Definition

Hard competition has forced companies to seek how to use scientific methods for decision making within the company, and also between the company and its suppliers and customers. The delay in the delivery of demands incurs a tardiness penalty due to customer dissatisfaction: possible contractual costs for late delivery with potential loss of reputation, also the demands which are finished before deadline and not delivered to a customer could result in additional storage or insurance costs, or even product deterioration. The setup time is the most important factor in production lines where it is possible large from the production run. Section 8 describes a genetic algorithm procedure for TSPPCA.

Aim of the Work

This work aims to determine optimum product sequencing for multiple demands with minimum setup times using a genetic algorithm to measure customer satisfaction in the manufacturing environment, which is characterised by its rapid change and development.

Decision Making (DM)

Operations decisions range from simple judgments to complex analyses, which also involves judgment. Judgment typically incorporates basic knowledge, experience, and common sense. They enable a blending of objectives and sub-objective data to arrive at a choice. The appropriateness of a given type of analysis depends on:

1. The significant or long-lasting decisions
2. The time availability and the cost of analysis
3. The degree of complexity of the decision.

An analytical and scientific framework for decision implies the following systematic steps:

a. Defining The Problem

Defining the problem enables identification of the relevant variables in the basis of the problem. A careful definition of the problem is crucial. Finding the root cause of a problem requires detailed questioning and detective work. If a problem definition is too narrow, the relevant variables may be omitted. If it is broader, many tangible aspects may be included which leads to complex relationships.

b. Establishing the Decision Criteria

Establishment the decision criteria is important because each criterion reflects the goals and purpose of the work efforts. For many years profits served as a convenient and accepted goal for many organisations, based on economic theory. Nowadays organisations will have multiple goals such as employee welfare, high productivity, stability, market share, growth, industrial leadership, and other social objectives.

c. Formulating the Model

The formulation of a model lies at the heart of the scientific decision making process. The model describes the essence of a problem or relationship by abstracting relevant variables from the real-world situation. Models are used to simplify or approximate reality, so the relationships can be expressed in tangible form and studied in isolation. Modelling a decision making situation usually requires both formulating a model and collecting the relevant data to use in the model. Mathematical and statistical models are the most useful models for understanding the complex business of the problem. Mathematical models can incorporate a factor that cannot readily be visualised. With the aid of computers and simulation techniques, these quantitative models are flexible.

d. Generating alternatives

Alternatives are generated by varying the values of the parameters. Mathematical and statistical models are particularly suitable for generating alternatives because they can be easily modified. The model builder can experiment with a model by substituting different values for the controllable and uncontrollable variables.

e. Evaluating the Alternatives

Evaluation of the alternatives is relatively objective in an analytical decision process because the criteria for evaluating the alternatives have been precisely defined. The best alternative is the one that most closely satisfies the criteria. Some models like the LPP model automatically seek out a maximizing or minimizing solution. In problems, various heuristic and statistical techniques can be used to suggest the best course of action.

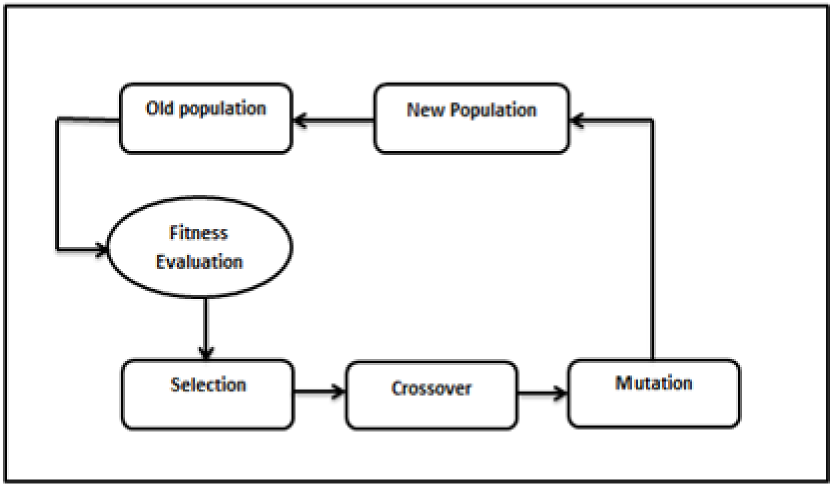
f. Implementing and Monitoring

Implementation and monitoring are essential for completing the managerial action. The best course of action or the solution to a problem determined through a model is implemented in the business world. Other managers have to be convinced of the merit of the solution. Then the follow-up procedures are required to ensure appropriate action is taken. This includes an analysis and evaluation of the solution along with the recommendations for changes or adjustments.

Genetic Algorithms (GA)

The GA operation is based on the Darwinian principle of survival of the fittest and it implies that the 'fitter' individuals are more likely to survive and have a greater chance of passing their 'good' genetic features to the next generation (Razali, 2014). Figure 1 illustrates the basic operation of GA. In the standard or basic procedure of GA (Larranaga and Kuijpers, 1999), an initial population is created containing a predefined number of individuals (i.e. solutions). Each individual has an associated fitness measure, typically representing an objective value. The concept that the fittest (or best) individuals in a population will produce fitter offspring is then implemented in order to reproduce the next population. Selected individuals are chosen for reproduction (by crossover and mutation) at each generation, with an appropriate crossover and mutation factor to randomly modify the genes of an individual. The algorithm identifies the individuals with optimising fitness values, and those with lower fitness will naturally get discarded from the population. Once crossover and mutation are done, a new generation is formed and the process is repeated until some stopping criteria have been reached (Youssef, 2001).

Figure 1. Basic Operation of Genetic Algorithm (Larranaga and Kuijpers, 1999)



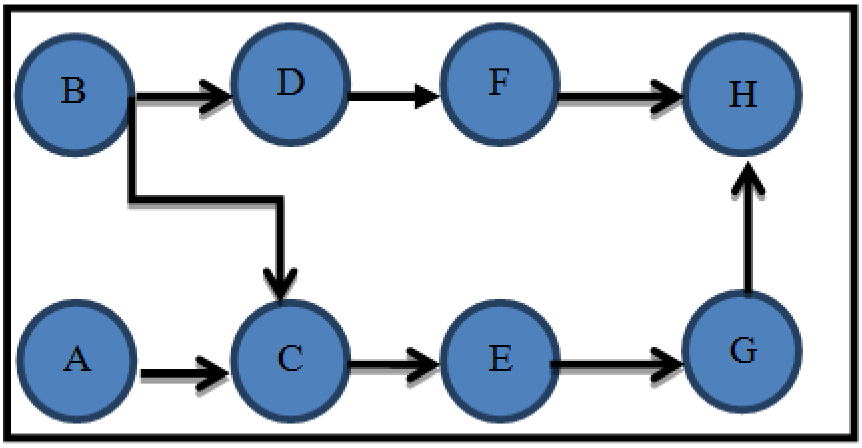
Travelling Salesman Problem with Precedence Constraints Approach

The basic Travelling Salesman Problem (TSP) has neither constraint nor priority given to any cities. The TSP with precedence constraints approach (TSPPCA) is one in which a set of n nodes and distances for each pair of nodes are given; the problem is to find a tour from node 1 to node n of minimal length which takes given precedence constraints into account. In TSPPCA some order of cities is given, and we ought to visit cities in that order only. TSPPCA differs from traditional TSP whereby in TSPPCA there is no need to return to the original product. TSPPCA becomes more important because in real-life problems we always have to follow some orders. An example of TSPPCA is shown in Figure 2. Each precedence constraint requires that some nodes have to be visited before some other node j (Kotecha and Gambhava, 2003). In a directed graph, the vertices (circles) represent activities or tasks and the edges represent the precedence relations between activities (Razali, 2014). The task dependencies deal with the relationships between giving tasks and how they affect each other. The four types of task dependencies are (Dreo et al., 2006):

1. Finish-to-start in which predecessor tasks must be finished before the successor can start.
2. Start-to-finish in which the successor task can finish only after the predecessor task has started.
3. Start-to-start in which two tasks can start simultaneously.
4. Finish-to-finish in which two tasks must finish at the same time.

In this paper, we deal with TSPPCA as Finish-to-start types of task dependencies, described above.

Figure 2. Example of TSP with Precedence Relationships (Kotecha and Gambhava, 2003)



Genetic Algorithm Procedure for TSPPCA

In TSPPCA, the precedence constraints require that certain nodes must precede certain other nodes in any feasible directed tour. For this reason, the use of a conventional genetic algorithm procedure for TSP, with an order-based representation, might generate invalid candidate solutions. The proposed TSPPC procedure maintains the main steps which are Initialisation and Representation, Evaluation and Selection and Generation of offspring as in the GA procedure for general TSP. The only difference is in the representation stage. The chromosomes in the initial population, as well as the offspring chromosomes created from the reproduction process, need to be repaired before going through the evaluation process. The procedure and the flowchart of the proposed algorithm are presented in Figure 3 and Figure 4 respectively.

Figure 3. Procedure of the proposed GA for TSPPCA

```
Procedure: Proposed algorithm
Begin
Step 1: Initialization & Representation
Step 1.1: Set GA parameters and the problem information
For i    1 to pop_sizedo
Step 1.2: Generate random permutation of sequence (x1,…xN) with N strings
End For
While number of generation <ngenerdo
While population <pop_sizedo
While length of chromosome <N  do
Step 1.3 Filter  Products Sequence
Step 1.3.1: Check and store available product within demand  in available set
Step 1.3.2: Select and store product in earlier position in updated sequence
Step 1.3.3: Remove product without demand from selected product
End While
End While
Step 2: Evaluation & Selection
Step 2.1: Calculate fitness value of each chromosome in i
Step 2.2: Select chromosomes in ias parents using Roulette Wheel selection scheme
Step 3: Generation of Offspring
Step 3.1: With probability Pc, select chromosomes in step 2.2 and apply linear order crossover
Step 3.2: With probability Pm, select chromosomes in step 3.1 and apply inversion mutation
End While
End procedure
```

There are several steps to determine the products sequence in multiple demands as following:

Step 1: Initialisation and Representation

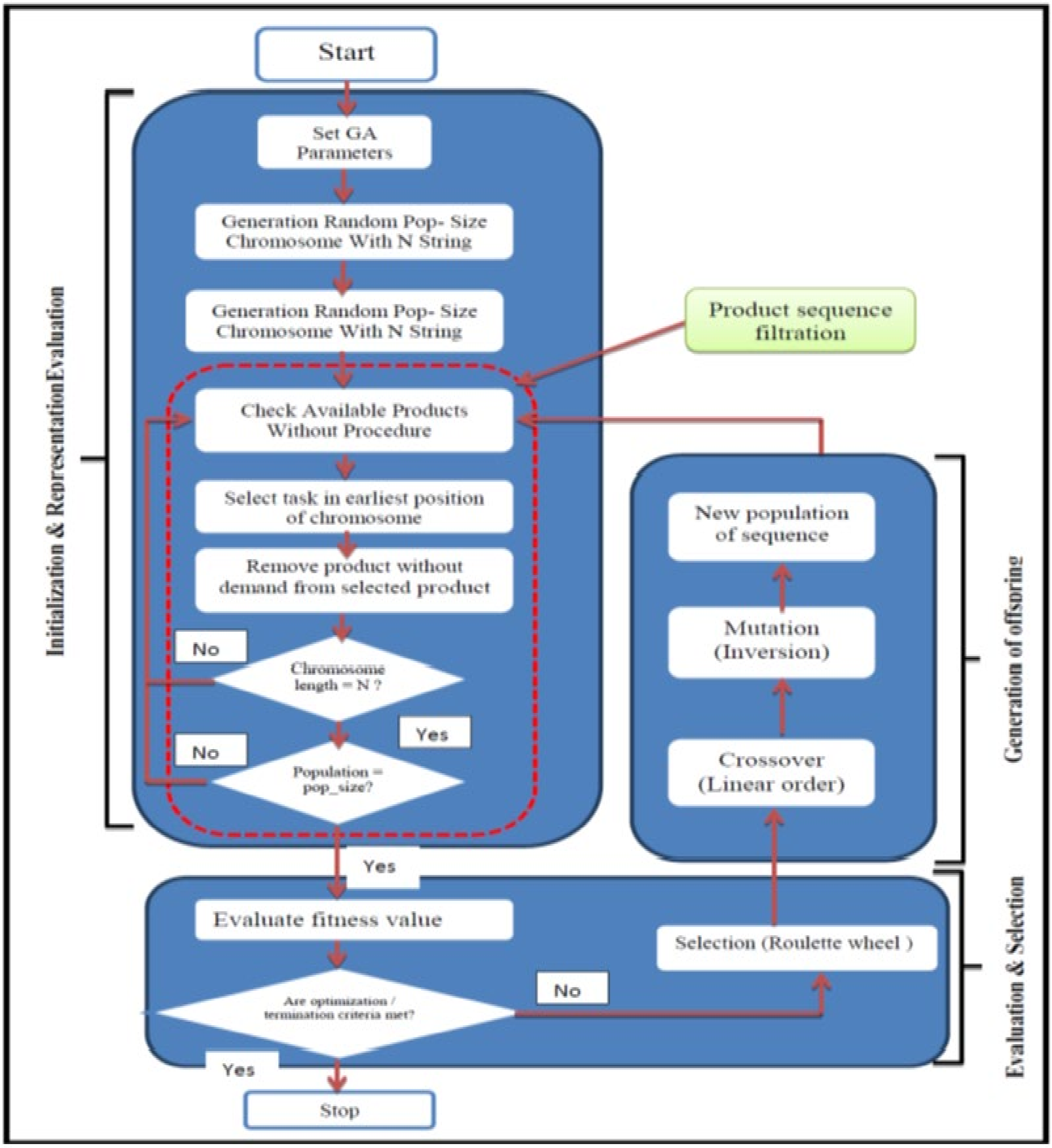
Step 1.1: Set GA parameters

The GA parameters such as population size, maximum number of generations, the probability of crossover and probability of mutation are set earlier in the program.

Step 1.2: Generate random initial population

In the initial population, the random permutation method is used to generate chromosomes. The integer from 1 to N, which is the number of products, is generated in random sequence. The number of chromosomes generated is depending on the size of population, pop size. These sequences normally did not satisfy the precedence constraint. Therefore, the infeasible chromosomes must be filtered using the topological sort technique.

Figure 4. Flowchart of the Proposed GA for TSP with Precedence Constraints



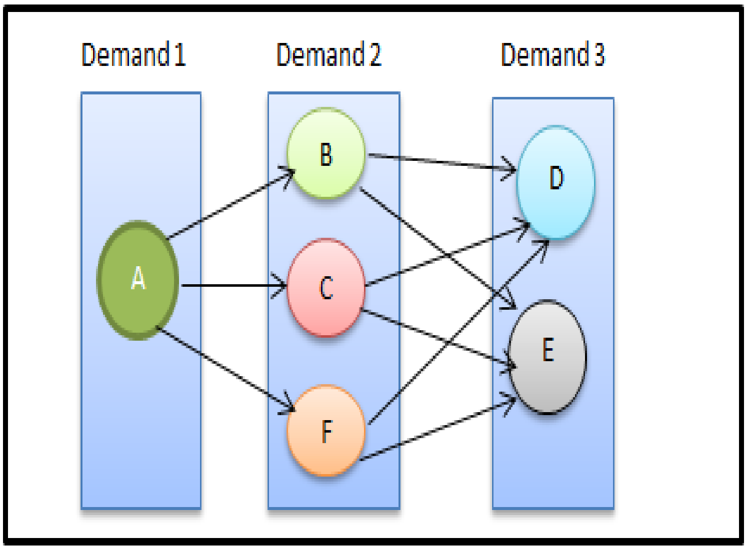
Step 1.3: Product sequence filtration

Product sequence filtration in the proposed algorithm for TSPPCA is based on the topological sort technique and is explained in Steps 1.3.1 through Steps 1.3.4.

Step 1.3.1: Check available product

Initially, a chromosome is generated randomly and may not feasible. For example, the chromosome structure represented as [DACFEB] is infeasible because it did not satisfy the precedence constraint to example consistent from demand three as shown in Figure 5. In order to filter the chromosomes to become feasible solutions, products without predecessor are selected and stored in available set. In this example, product A is the only task without predecessor and therefore product A is selected and being stored in sequence. Then the outgoing edges of product A, which is product B, C and F, are removed . As a result, the new available set consists of Product A, B and F as displayed in Table (1).

Figure 5. Example Diagram for Three Demands



Step 1.3.2: Select task in earliest position on chromosome

The selection of product to be stored in sequence is based on the "earliest position" found in the chromosome. By referring to the available set [B C F], Product C is firstly found in the chromosome [D A C F E B]. Therefore, Product C is selected as the second string to be stored in sequence and the updated sequence now consists of [A C].

***Step 1.3.3: Remove products has no relationship between products inside demand from selected products*.**

When product [C] is selected to be stored in sequence, the product has no relationship between products inside demand from selected products to be removed. Therefore, the relationships ($\rightarrow$ D, C $\rightarrow$ E) are removed, and the new available set is consisting of [B, F]. Again, based on 'earliest position' selection of task approach, task F first appeared before product B in the chromosome [DACFEB] and therefore product F is selected to be placed in updated sequence. The selection procedure is repeated until the length of the sequence is equal to N. The final feasible path that is generated from this approach is [A C F B D E].

Step 2: Evaluation and Selection

A mechanism to select an individual in the population for reproduction to create new offspring or to transfer a part of the existing population to the next generation is needed. It is possible to perform the task of selection completely in a randomised fashion. This selection mechanism will eventually cause the algorithm to reach global minimum/maximum. However, using this scheme, convergence of the population will almost be impossible, and termination will take a considerably long time (Razali, 2014). The selection strategy addresses which of the chromosomes in the current generation will be used to reproduce offspring in hopes that the next generation will have even higher fitness. A number of selection techniques exist including elitist, tournament (Dreo et al., 2006).

Step 2.1: Calculate fitness value

The fitness value of each chromosome in the population is evaluated using the fitness function in Equation (1).

Minimise setup time= $\sum_{i=1}^{N} \sum_{j=1}^{N} t_{ij} X_{ij}$ (1)

Where:

Xij: Decisions variable for From-To Matrix $= \begin{cases} 1 & \text{when selected relation} \\ 0 & \text{Otherwise} \end{cases}$

tij: represents the setup time between product i to product j.

Step 2.2: Parent selection

The Roulette Wheel selection is used to select parent chromosomes to be re-generated for the next chromosome. By using the proportional Roulette Wheel, all individuals are given a chance to be selected and the chances of the fitter individual to be selected as a parent for crossover are higher.

Step 3: Generation of offspring

Reproduction is the crossover of two chromosomes to produce a new offspring that has genes from both parents. In nature, although it may be much more complicated, crossover basically occurs as follows: chromosomes of both parents are randomly divided from the same gene positions into a number of segments and the corresponding segments are exchanged and copied to the chromosome of the newly created offspring. Therefore, the offspring inherits traits from both parents (Razali, 2014). In the genetic algorithm, special techniques for permutation-based chromosomes are deployed, which ensure that, when applied on two permutation-based chromosomes, the chromosomes of the resulting offspring are also valid permutations.

Step 3.1: Crossover

Linear order crossover is used to generate two new offspring. This operator is the most frequently used for the crossover operation when the chromosome representation is ordinal (Mokhlesian, 2010). This crossover operator can preserve both the relative positions between genes and the absolute positions relative to the extremities of parents as much as possible.

Step 3.2: Mutation

Mutation operation based on inversion (flip) as described in the GA procedure for TSP is applied in the chromosome after the crossover process. In order to avoid getting stuck onto a local minimum and to avoid premature convergence, population diversity is required to be kept up to some extent. In the genetic algorithm, this is achieved by the help of a mutation mechanism, which causes some sudden changes to the traits of individuals according to a predefined mutation probability parameter. A new offspring can be achieved in different ways either by flipping, inserting, swapping or sliding the allele values at two randomly chosen gene positions. The inversion mutation (flipping) operator (Razali, 2014) randomly selects two cut points in the chromosome, and it reverses the sub tour between these two cut points.

Case Study

Wires Factory in the Ur Company is one of the factories that produce semi-finished products that serve some of the company's labs as well as external demand such as Diyala for electrical products. The main products of the factory are as follows:

1. Wire with a single-layer or two-layer insulated loop used in coil electric motors, manufacture of electrical transformers and measurements (0.25-3.5 mm), produced under (DIN4635 - IEC317) and represented as symbol (A).
2. Wires with a rectangular section insulated with a layer or several layers of paper shall be produced according to the international specifications (JIS3104) with measurements (14 x 3.5) - (6 x 0.2) mm and represented as symbol (B).
3. Wire with rectangular section insulated with amine is produced according to British Standard (BS4516) and represented as symbol C.
4. Wire winding with a rectangular section insulated with double-pole paper is produced according to specification (JIS3104) and represented as symbol (D).

These products which will be made on the same production line inside the factory and need setup time to change arrangements on this production line when altering the production to another product can be illustrated as setup time for the product's factory in Table 1. The planning department orders a monthly set of demands as illustrated in Table 2.

Table 1: Setup time (hours): from-to matrix

	To product j				
From product i	Products Products	A	B	C	D
	A	0	91	102	210
	B	50	0	80	110
	C	20	35	0	70
	D	30	40	65	0

Table 2: Sequence of demands depend on order from the planning department

Months	Products for each month
January	A, B, C, D
February	A, B, C, D
March	B, C
April	A, B, D
May	A, B, C, D
June	C, D
July	A, B
August	A, D
September	A, B, C, D
October	A
November	B, C, D
December	A, B, C, D

The Results of Product Sequencing Problem Using GA

Products sequence for the wires factory when customers order multiple demands with several products could be solved by GA (using MATLAB program) where this approach takes into account the product which ended the previous demand, and which will begin in the later demand. GA consists of two matrices, where the first matrix is the constraints matrix, which represents the relationships constraints between the demand's products relying on an opinion of the planning department for a period of twelve months: this consists of twelve demands in thirty six rows, as shown in Table 3. The first row of this matrix represents the start of the sequence of products which begins at ending the previous demand and its product A, while the remainder from the rows represent products of this demand. The second matrix is a setup time matrix as shown in Table 4, where this matrix is a square matrix and consists of thirty-six rows and columns which represent products of this demand and is added to a product which ends the previous demand.

Table 3: Constraints Matrix for Order Planning Department

Months	Product symbol	No. product	The relationships correlation between products						
product that ended the previous demand	A	1	0	0	0	0	0	0	0
January	A	2	0	1	0	0	0	0	0
	B	3	0	1	0	0	0	0	0
	C	4	0	1	0	0	0	0	0
	D	5	0	1	0	0	0	0	0
February	A	6	0	2	3	4	5	0	0
	B	7	0	2	3	4	5	0	0
	C	8	0	2	3	4	5	0	0
	D	9	0	2	3	4	5	0	0
March	B	10	0	6	7	8	9	0	0
	C	11	0	6	7	8	9	0	0
April	A	12	0	10	11	0	0	0	0
	B	13	0	10	11	0	0	0	0
	D	14	0	10	11	0	0	0	0
May	A	15	0	12	13	14	0	0	0
	B	16	0	12	13	14	0	0	0
	C	17	0	12	13	14	0	0	0
	D	18	0	12	13	14	0	0	0
June	C	19	0	15	16	17	18	0	0
	D	20	0	15	16	17	18	0	0
July	A	21	0	19	20	0	0	0	0
	B	22	0	19	20	0	0	0	0
August	A	23	0	21	22	0	0	0	0
	D	24	0	21	22	0	0	0	0
September	A	25	0	23	24	0	0	0	0
	B	26	0	23	24	0	0	0	0
	C	27	0	23	24	0	0	0	0
	D	28	0	23	24	0	0	0	0
October	A	29	0	25	26	27	28	0	0
November	B	30	0	29	0	0	0	0	0
	C	31	0	29	0	0	0	0	0
	D	32	0	29	0	0	0	0	0
December	A	33	0	30	31	32	0	0	0
	B	34	0	30	31	32	0	0	0
	C	35	0	30	31	32	0	0	0
	D	36	0	30	31	32	0	0	0

Table 4: Setup Matrix for Order Planning Department

To product j																																						From product i
December				November			October	September				August		July		June		May				April			March		February				January				-	Months		
D	C	B	A	D	C	B	A	D	C	B	A	D	A	B	A	D	C	D	C	B	A	D	B	A	C	B	D	C	B	A	D	C	B	A	A	Products \ Products	Months	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	210	102	91	0	0	A	-	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	210	102	91	0	210	102	91	0	1000	A	January	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	110	80	0	50	110	80	0	50	1000	B		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	70	0	35	20	70	0	35	20	1000	C		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	65	40	30	0	65	40	30	1000	D		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	102	91	210	102	91	0	0	0	0	0	0	A	February	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	80	0	110	80	0	50	0	0	0	0	0	B		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	35	70	0	35	20	0	0	0	0	0	C		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	65	40	0	65	40	30	0	0	0	0	0	D		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	80	0	0	0	0	0	0	0	0	0	0	B	March	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	35	0	0	0	0	0	0	0	0	0	C		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	210	102	91	0	210	91	0	0	0	0	0	0	0	0	0	0	0	0	A	April	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	110	80	0	50	110	0	50	0	0	0	0	0	0	0	0	0	0	0	B		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	65	40	30	0	40	30	0	0	0	0	0	0	0	0	0	0	0	D		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	210	102	210	102	91	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	A	May	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	110	80	110	80	0	50	0	0	0	0	0	0	0	0	0	0	0	0	0	0	B		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	70	0	70	0	35	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	C		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	65	0	65	40	30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	D		

0	0	0	0	0	0	0	0	0	0	0	0	0	0	35	20	70	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	C	June
0	0	0	0	0	0	0	0	0	0	0	0	0	0	40	30	0	65	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	D	
0	0	0	0	0	0	0	0	0	0	0	0	210	0	91	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	A	July
0	0	0	0	0	0	0	0	0	0	0	0	110	50	0	50	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	B	
0	0	0	0	0	0	0	0	102	91	0	210	210	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	A	August
0	0	0	0	0	0	0	0	65	40	30	0	0	30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	D	
0	0	0	0	0	0	0	0	210	102	91	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	A	September
0	0	0	0	0	0	0	50	110	80	0	50	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	B	
0	0	0	0	0	0	0	20	70	0	35	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	C	
0	0	0	0	0	0	0	30	0	65	40	30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	D	
0	0	0	0	210	102	91	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	A	October
110	80	0	50	110	80	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	B	November
70	0	35	20	70	0	35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	C	
0	65	40	30	0	65	40	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	D	
210	102	91	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	A	December
110	80	0	50	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	B	
70	0	35	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	C	
0	65	40	30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	D	

Figure 6. shows a setup time and a number of generations. Forty-four generations represents the optimal sequence for these demands with minimum setup time, which comes to 1730 hours. Figure 7. shows the optimum products sequencing for these demands and elapsed time to run this program which was 52 seconds.

Figure 6. Setup Time and Number of Generation GA

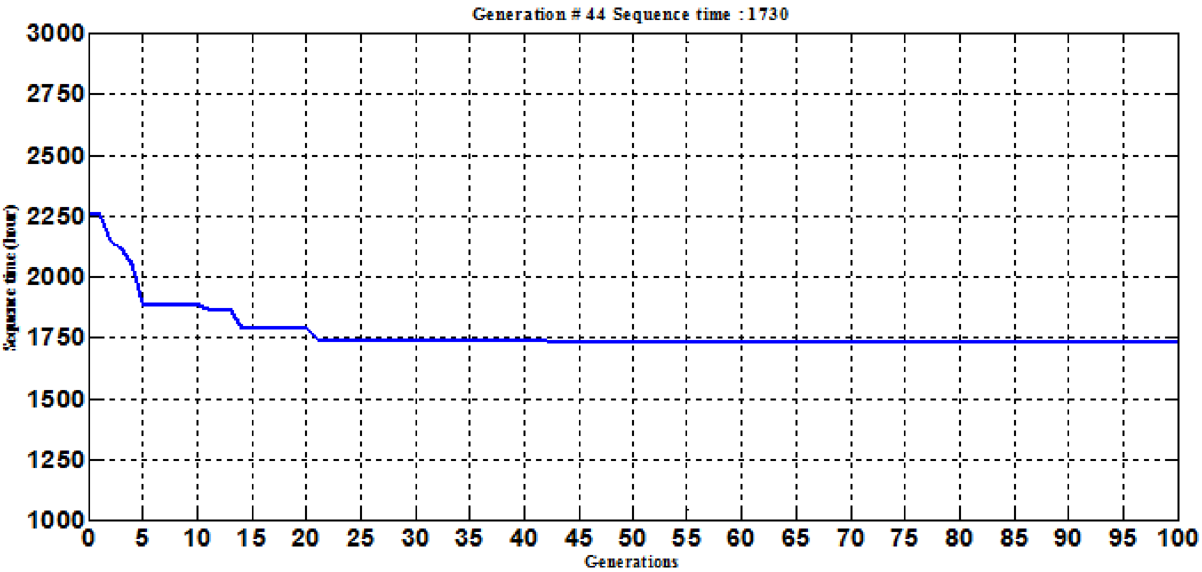
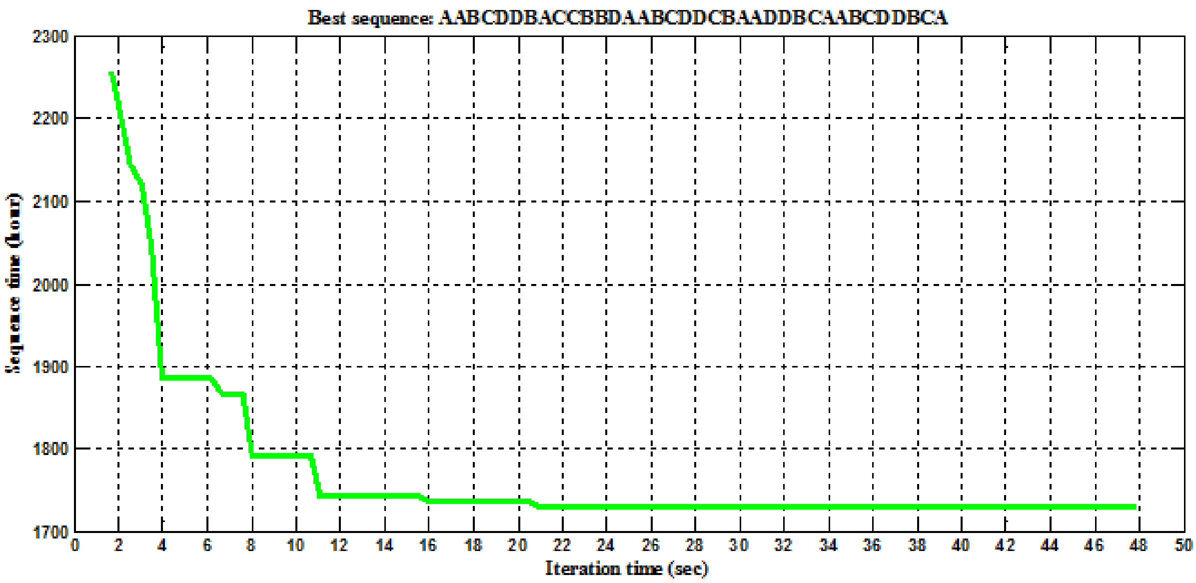


Figure 7. Optimal sequencing for products



The optimum sequence for this demand using this method as follows:
A ABCD DBAC CB BDA ABCD DC BA AD DBCA A BCD DBCA

Conclusions and Future Works

Through finding the optimal sequence for products of the wire factory, we conclude the following:

1. The use of quantitative methods helps decision makers in productive companies solve the problems of sequence for products.
2.Genetic Algorithm (GA) is an efficient technical method to solve product sequence problems, especially when product ranges are large.
3. GA takes into account the product which ended the previous demand, and which will begin in the later demand.
4. GA is an advantage in long-term planning to determine a sequence of products.

The future applications of our paper can be summarised as follows.

1- Direct the attention of decision makers in factories and production companies to apply modern quantitative methods.
2- Cooperation between universities and the industrial sector, through the formation of joint working teams of different disciplines, will allow direct access to the most prominent problems.
3- Follow-up the actual application of studies and results obtained from the joint working teams in the industrial sector.

REFERENCES

Anil, S.K. & Suresh, N. (2009). Operation management. New age international Daryaganj, New Delhi, 2(63): 136 – 147.

Charnprasitphon, A. (2007). Modelling and analysis of the batch production scheduling problem for perishable products with setup times. PhD Thesis, School of Industrial and System Engineering, Georgia Institute of Technology, 6 (49): 252 – 272.

Cheng, T., Liu, Z. & Yu, W. (2001). Scheduling jobs with release dates and deadlines on a batch processing machine. IIE Transactions, 33(72): 163 - 183.

Clark, A., Morabito, R. & Toso, E. (2010). Production setup sequencing and lot-sizing at an animal nutrition plant through ATSP subtour elimination and patching. Journal of Scheduling, 13(41): 161 – 185.

Costa, A., Fichera, S. & Cappadonna, F. (2013). A genetic algorithm for scheduling both job families and skilled workforce. Int. J. Oper. Quant. Manag., 19(85): 146 – 158.

Dreo, A. P. J., Siarry, P. & Taillard, E. (2006). Metaheuristics for hard optimisation; methods and case studies. Springer Berlin Heidelberg, 5(72): 271- 283.

Jingxu, H. (2008). Heuristic procedures to solve sequencing and scheduling problems in automobile industry. Ph.D Thesis./ The University of Tennessee, Knoxville, 7(44): 211 – 241.

Kotecha, K. & Gambhava, N. (2003). Solving precedence constraint travelling salesman problem using genetic algorithm. National Conference on Software Agents, 6(21): 114 – 143.

Kreipl, S. & Pinedo, M. (2004). Planning and scheduling in supply chains: An overview of issues in practice. Production and Operations Management Society, Spring, 13(39): 162 – 178.

Laith, W. H., Abed, A. S. S. & Mahmoud, A. M. (2015). Determine the optimal sequence - dependent setup cost and / or setup time for single demand with multiple products using modified assignment method. Industrial Engineering Letters, 5(8): 161 - 195.

Larranaga, P. & Kuijpers, C. (1999). Genetic algorithms for the travelling salesman problem: A review of representation and operators. Artificial Intelligence Review, 13(74): 129-170.

Mohammadi, G. & Mohammadi, D. (2016). Scheduling a flow line manufacturing cell with sequence dependent family setup times using bilateral genetic algorithm (BGA). International Journal of Service Science, Management and Engineering, 3(3): 145 – 167.

Mokhlesian, M. (2010). economic lot scheduling problem with consideration of money time value. International Journal of Industrial Engineering Computations, 1(47): 121-138.

Patricia, C. (2014). Shop scheduling in the presence of batching, sequence-dependent setups and incompatible job families minimising earliness and tardiness penalties. PhD. Thesis, College of Engineering and Computer Science University of Central Florida, 1(31): 115 – 146.

Pinedo, M., (2002). Scheduling: Theory, algorithms and systems. 2nd ed, Prentice Hall, Englewood Cliffs, NJ, 1(24): 85 – 101.

Razali, N. (2014). Genetic algorithm for process sequencing modelled as the travelling salesman problem with precedence constraints. PhD. Thesis, Dublin City University, 3(15): 211 – 224.

Riddalls, C.E. & Bennett, S. (2001). The optimal control of batched production and its effect on demand amplification. International Journal of Production Economics, 6(72): 125 – 137.

Salmasi, N., Logendran, R. & Skandari, M. (2011). Makespan minimisation of a flowshop sequence dependent group scheduling problem. Int. J. Adv. Manuf. Technol. 2(56): 151 - 170.

Youssef, H. (2001). Evolutionary algorithms, simulated annealing and tabu search: A comparative study. Engineering Applications of Artificial Intelligence, 5(14): 167–181.

Zhu, X. & Wilhelm, W.E. (2006). Scheduling and lot sizing with sequence-dependent setup: A literature review. IIE Transactions, 4(38), pp. 987-1007.