International Journal of Advanced Research Methodology in Engineering & Technology,
Volume 1, Issue1, December 2016

Inventory Optimization: A Meta Heuristics Approach

P.Mathiyalagan [1] P.Parthiban[2] K.Tamil Mannan[3]

[1,] Dept of Mechanical Engineering, Galgotias University, Greater Noida, Uttar Pradesh
(1) mathis09051970@yahoo.co.in

[2] Dept of Production Engineering, National Institute of Technology Trichy,
(2) parthee_p@yahoo.com

[3] Dept of Mechanical Engineering, Indira Gandhi National Open University
(3) ktmannan@radiffmail.com

ABSTRACT

Inventory management plays a vital role in supply chain management. The service provided to the customer eventually gets enhanced once the efficient and effective management of inventory is carried out all through the supply chain. Thus the determination of the inventory to be held at various levels in a supply chain becomes inevitable so as to ensure minimal cost for the supply chain. Minimizing the total supply chain cost is meant for minimizing holding and shortage cost in the entire supply chain as well as the lead times at various stages. A serious issue in the implementation of the same is that the excess stock level and shortage level is not static for every period. In this report, study has been carried out on Genetic Algorithms and Particle Swarm Optimization in order to distinctively determine the most probable excess stock level and shortage level required for inventory optimization in the supply chain such that the total supply chain cost is minimized.

Key Words: Particle Swarm Optimization, Genetic algorithm.

1. INTRODUCTION

The concepts of supply chain management incorporates a wide range of activities that support the planning, implementation and control manufacturing and the delivery processes right from the source of raw material to the spot where the end product is utilized. Acute issues in supply chain management arise out of shorter product lifecycles that lead to higher demand uncertainty and the action on global markets consequently increasing the supply chain complexity. From the operational point of view, this research addresses four problem areas including Inventory management and control; production, planning and scheduling; information sharing, coordination, monitoring; and operation tools.

A collection of items held by an organization for future utilization is known as inventory. Inventory optimization application aids in the enhancement of inventory control and its management across an extended supply network, which organizes the latest techniques and technologies. Inventory control describes the design and management of the storage policies and procedures for raw materials, work-in-process inventories, and usually, final products. Estimation of the precise amount of inventory at each point in the supply chain devoid of excesses and shortages despite minimizing the total supply chain cost is a chief concern for the inventory and supply chain managers. The precise estimation of optimal inventory is essential since shortage of inventory yields to lost sales, while excess of inventory may result in pointless storage costs.

Genetic algorithm and Particle swarm optimization can be used for inventory optimization in different stages of the supply chain. It determines the most possible excess stock level and shortage level that is needed for inventory optimization in the supply chain so as to minimize the total supply chain cost. The practical problem that is arising usually in inventory management is the dynamic nature of the excess stock level and shortage level over all the periods. The necessary operation to do is to determine the stock level that occurs in a maximum rate. So, the optimization will be effective only if the maximum occurrences of stock level are considered.

2.LITERATURE REVIEW

Radhakrishnan et al [1] developed a new and efficient approach that works on Genetic Algorithms to minimize the total supply chain cost. K.Balaji et al [2] utilized an Analytic Hierarchy Process (AHP) for estimating the value of the inventory system using Multi Criteria Inventory Classification (MCIC). Birger Raa [3] carried out an analysis of cyclic versus reactive planning for inventory routing. Wikrom Jaruphongsa et al [4] studied to find an integrated replenishment policy for the Third Party Warehouse and Distribution Centre simultaneously to satisfy all demands at the DC at minimum cost. Tarun Kumar et al [5] used the application of Genetic Algorithm to a three stage, six member supply chain for inventory optimization. Chingping Han et al [6] carried out stochastic modelling of a two-echelon multiple sourcing supply chain system with genetic algorithm. N.Jeyanthi et al [7] applied Genetic Algorithm to Inventory Optimization of a Supply Chain with four members. Amine Ayad [8] optimized inventory and store results in big box retail environment. A fresh genetic algorithm (GA) approach for the integrated inventory distribution problem (IIDP) has been projected by Abdel et al. [9]. They have developed a genetic representation and have utilized a randomized version of a formerly developed construction heuristic in order to produce the initial random population. In [10] Pupong et al., have put forth an optimization tool that works on basis of a multi-matrix real-coded Generic Algorithm (MRGA) and aids in reduction of total costs associated with supply chain logistics. They have incorporated procedures that ensure feasible solutions such as the chromosome initialization procedure, crossover and mutation operations. They have evaluated the algorithm with the aid of three sizes of benchmarking dataset of logistic chain network that are conventionally faced by most global manufacturing companies.

In 1995, Kennedy and Eberhartin, inspired by the choreography of a bird flock, first proposed the Particle Swarm Optimization (PSO). In comparison with the evolutionary algorithm, PSO, a relatively recently devised population-based stochastic global optimization algorithm has many similarities and the robust performance of the proposed method over a variety of difficult optimization problems has been proved [11]. In PSO, the potential solutions, called particles follow the current optimum particles to fly through the problem space. Every particle represents a candidate solution to the optimization problem. The best position visited by the particle and the position of the best particle in the particle's neighbourhood influence its position. The ability of the particles to remember the best position that they have seen is an advantage of PSO. An evaluation function that is to be optimized evaluates the fitness values of all the particles [12].

3.INVENTORY OPTIMIZATION USING GENETIC ALGORITHM

The method uses the Genetic Algorithm to study the stock level that needs essential inventory control. In practice, the supply chain is of length *n*, means having *n* number of members in supply chain such as factory, distribution centers, suppliers, retailers and so on. Here, for instance we are going to use a three stage supply chain that is illustrated in the figure 1. Our exemplary supply chain consists of a factory, distribution center1 and distribution centre 2.

Figure 1. Three member supply chain

In the supply chain we are illustrated, the factory is the massive stock holding area where the stocks are manufactured as per the requirement of the distribution center 1. Then the distribution center 1 will take care of the stock to be supplied for the distribution cen 2. The responsibility of our approach is to predict an optimum stock level by using the past records and so that by using the predicted stock level there will be no excess amount of stocks and also there is less means for any shortage. Hence it can be asserted that this approach eventually gives the amount of stock levels that needs to be held in the three members of the supply chain, factory, distribution center1 and distribution center 2.

In this methodology, we are using genetic algorithm for finding the optimal value. The flow of operation of our methodology is clearly illustrated in figure 2 which depicts the steps applied for the optimization analysis.

Initially, the amount of stock levels that are in excess and the amount of stocks in shortage in the different supply chain contributors are represented by zero or non-zero values. Zero refers that the contributor needs no inventory control while the non-zero data requires the inventory control. The excess amount is given as positive value and the shortage amount is mentioned as negative value.

The first process needs to do is the clustering that clusters the stock levels that are either in excess or in shortage and the stock levels that are neither in excess nor in shortage separately. This is done simply by clustering the zero and non-zero values. For this purpose we are using, the efficient K-means Clustering algorithm.

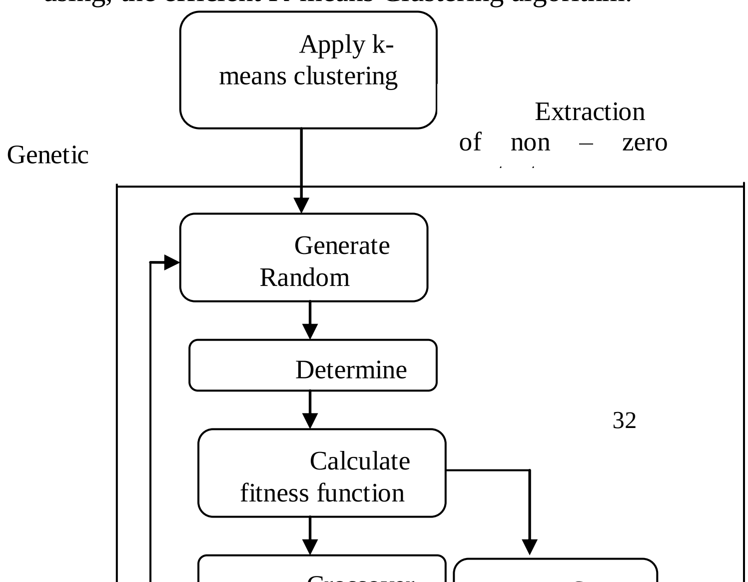

Figure 2. Operational flow of proposed methodology

After the process of K- means clustering is performed, the work starts its proceedings on Genetic algorithm, the heart of our work.

3.1. Chromosome Representation

The randomly generated initial chromosome is created by having the stock levels within the lower limit and the upper limit for all the contributors of the supply chain, factory and the distribution centers. The stock level of each member of the chromosome is referred as gene of the chromosome. Hence for *n* length supply chain, the chromosome length is also *n* . As we are using only three members of the chain, the length of the chromosome *n* is 3, i.e. 3 genes. And the chromosome representation is pictured in figure 3. Each gene of the chromosome is representing the amount of stock that is in excess or in shortage.

Chromosome 1 Chromosome 2

-300	800	-400	100	-600	900

Figure 3. Chromosome representation

These kinds of chromosomes are generated for the genetic operation. Initially, only two chromosomes will be generated and from the next generation a single random chromosome value will be generated. The chromosomes thus generated is then applied to find its number of occurrences in the database content by using a *Select count*( ) function.

The function will give the number of occurrences of the particular amount of stock level for the three members *Nc* that are going to be used further in the fitness function.

3.2. Fitness Function

Fitness functions ensure that the evolution is toward optimization by calculating the fitness value for each individual in the population. The fitness value evaluates the performance of each individual in the population.

$$f(k) = log\left(1 - \frac{N_c}{N_p}\right), k = 1,2,3 \ldots.m$$

Where, N_c is the number of counts that occurs throughout the period, N_p is the total number of inventory values obtained after clustering, *m* is the total number of chromosomes for which the fitness function is carried out for each chromosome and the chromosomes are sorted on the basis of the result of the fitness function. Then the chromosomes are subjected for the genetic operation crossover and mutation.

3.3. Crossover

As far as the crossover operation is concerned, a single point crossover operator is used in this study. The first two chromosomes in the mating pool are selected for crossover operation. The crossover operation that is performed for an exemplary case is shown in the following figure.

Before Crossover

Cp ↓

-500	-700	-400

Cp ↓

After Crossover

Cp ↓

Cp ↓

Figure 4. Crossover

3.4. Mutation

By performing the mutation, a new chromosome will be generated. This is done by a random generation of two points and then performing swaps between both the genes. The illustration of mutation operation is shown below.

Before Mutation

-500	-800	300

Before Mutation

-500	300	-800

Figure 5. Mutation

The mutation operation provides new chromosomes that do not resemble the initially generated chromosomes. After obtaining the new chromosome, another random chromosome will be generated. Then again the process repeats for a particular number of iteration while the two chromosomes that are going to be subjected for the process is decided by the result of the fitness function. Each number of iteration will give a best chromosome and this is will be considered to find an optimal solution for the inventory control. When the number of iterations is increased then the obtained solution moves very closer to the accurate solution. More the number of iterations results in more accurate optimal solution.

3.5 Result

100	-800	300

-500	800	-300

100	700	-400

The two initial chromosomes are generated at the beginning of the genetic algorithm are '-546 -802 209' and '-155 248 - 759'. These initial chromosomes are subjected for the genetic operators, Crossover and Mutation. The resultant chromosome thus obtained after the application of crossover and mutation is '546 -759 248'. As for our iteration value of '100', the resultant chromosome moved towards the best chromosome after the each iterative execution. Hence at the end of the execution of 100th iteration, best chromosome '-591 -329 269' is obtained. While comparing the obtained result from the genetic algorithm with the past records, it can be decided that controlling this resultant chromosome is sufficient to reduce the loss either due to the holding of excess stocks or due to the shortage of stocks. Hence it is proved that the analysis obtains a stock level that is a better prediction for the inventory optimization in supply chain management.

4. Inventory Optimization using Particle Swarm Optimization (PSO)

The particle Swarm Optimization (PSO) is utilized to predict the optimal stock levels to be maintained in the future to minimize the supply chain cost. The procedures involved in determining the optimal stock levels are illustrated in figure 6 and the methodology is outlined below.

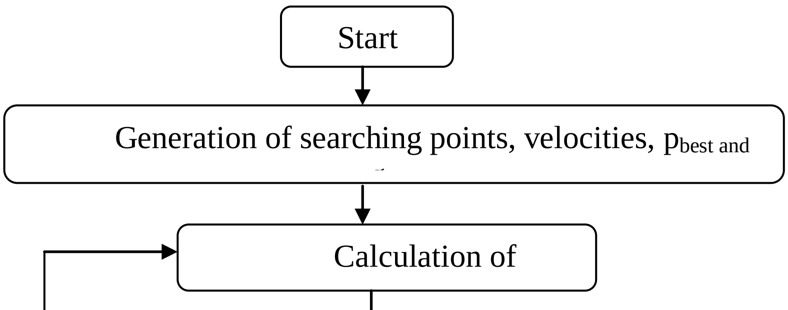

Figure6. Particle swarm optimization in optimizing the Base stock levels

The individuals of the population including searching points, velocities, p_{best} and g_{best} are initialized randomly but within the lower and upper bounds of the stock levels for all supply chain members, which have to be specified in advance.

4.1 Determination of Evaluation function

$$f(i) = w_1 \; log\left(1 - \frac{n_{occ}(i)}{n_{tot}}\right) + log(w_2 . t_{stock})$$

n_{occ} (i) is the number of occurrences of the particle in the record set, n_{tot} is the total number of records that have been collected from the past/ total number of data present in the record set, n is the total number of particles for which the fitness function is calculated, w_1 and w_2 are the weightings of the factors, stock levels, lead time of stocks in optimization, respectively and they are determined as:

$$w_1 = \frac{R_1}{R_1 + R_2}$$

$$w_2 = \frac{R_2}{R_1 + R_2}$$

R_1 and R_2 are the priority levels of influence of stock levels and lead time of stocks in optimization of respectively. Increasing the priority level of a factor increases the influence of the corresponding factor in the evaluation function. Hence this R_1 and R_2 decide the amount of influence of the factors The lead time of the stocks t_{stock} is determined as follows:

$$t_{stock} = \sum_{i=1}^{l-1} \sum_{q} t_{q,i}$$

For every individual, a comparison is made between its evaluation value and its p_{best} .The g_{best} indicates the most excellent evaluation value among the p_{best} . This g_{best} nothing but an index that points the best individual we have generated so far.

Subsequently the adjustment of the velocity of each particle a is as follows:

$$v_{new}(a,b) = w^{*} v_{cnt}(a) + c_1 * r_1 * [p_{best}(a,b) - I_{cnt}(a,b)] + c_2{}^{*} r_2 * [g_{best}(b) - I_{cnt}(a,b)]$$

Where,

a = 1,2, Λ, Λ, N_p

b= 1,2, $\Lambda, \Lambda, , d$

Here, v_{cnt} (a) represents current velocity of the particle, v_{new} (a,b) represents new velocity of a particular parameter of a particle, r_1 and r_2 are arbitrary numbers in the interval [0,1] , c_1 and c_2 are acceleration constants (often chosen as 2.0), w is the inertia weight that is given as:

$$w = w_{max} - \frac{w_{max} - w_{min}}{iter_{max}} \times iter$$

w_{max} and w_{min} are the maximum and minimum inertia weight factors respectively that are chosen randomly in the interval [0,1], $iter_{max}$ is the maximum number of iterations, *iter* is the current number of iteration.

Such newly obtained particle should not exceed the limits. This would be checked and corrected before proceeding further as follows,

If $v_{new}(a,b) > v_{max}(b)$, then $v_{new}(a,b) = v_{max}(b)$

if $v_{new}(a,b) < v_{min}(b)$, then $v_{new}(a,b) = v_{min}(b)$

Then, as per the newly obtained velocity, the parameters of each particle is changed as follows

$$I_{new}(a,b) = I_{cnt}(a,b) + v_{new}(a,b)$$

Then the parameter of each particle is also verified whether it is beyond the lower bound and upper bound limits. If the parameter is lower than the corresponding lower bound limit then replace the new parameter by the lower bound value. If the parameter is higher than the corresponding upper bound value, then replace the new parameter by the upper bound value. For instance,

If $P_k < P_{L.B,}$ then $P_k = P_{L.B}$

Similarly, if $P_k > P_{U.B,}$ then $P_k = P_{U.B}$

This is to be done for the other parameters also.

This process will be repeated again and again until the evaluation function value is stabilizing and the algorithm has converged towards optimal solution. The latest g_{best} pointing the individual is the best individual which is having the stock levels that are to be considered and by taking necessary steps to eliminate the identified emerging excesses/ shortages at different members of the supply chain, near optimal inventory levels can be maintained and the supply chain cost can be minimized to that extent.

4.2 Implementation

We have implemented the analysis based on PSO for optimal inventory control in the platform of MATLAB. As stated, we have the detailed information about the excess and the shortage stock levels in each supply chain member, the lead times of product stock levels to replenish each supply chain member as well as raw material lead time. The sample data having this information is given in the Table 1.

Table 1: A sample data set along with its stock levels in each member of the supply chain

I	F	D1	D2
	232	424	247
	-415	488	-912
	369	-686	-468
	459	289	-522
	-663	944	856
	-768	-937	-768

The Table 1 is having the Transportation ID, the stock levels which are in excess or in shortage at each supply chain member. Negative values represent shortage of stock levels and positive values represent the excess of stock levels.

As initialization step of the PSO process, the random individuals and their corresponding velocities are generated.

Table 2: Initial random individuals

F	D1	D2
255	61	215
354	-154	145

Table 3: Initial Random velocities corresponding to each particle of the individual

F	D1	D2
0.1298	0.1298	0.1298
0.0376	0.0376	0.0376

The simulation run on a huge database of 5000 past records showing evaluation function improvement at different levels of iteration is as follows:

Simulation Result showing evaluation function improvement with $w_1 = 0.6250$; $w_2 = 0.375$

For iteration 50: evaluation function = 5.6845;

For iteration 60; evaluation function = 5.5450; Improvement: 2%

For iteration 70; evaluation function = 5.4749; Improvement: 5%

For iteration 100; evaluation function = 4.7220; Improvement: 10%

As for deciding the total number of iterations required, the criteria followed is that as long as minimization of the Evaluation function is still possible, then the iteration continues till such a time that no improvement in the Evaluation function value is noticeable. After a certain number of iterations, if the evaluation function value is not improving from the previous iterations, then this is an indication that the evaluation function value is stabilizing and the algorithm has converged towards optimal solution. For greater accuracy, the number of iterations should be sufficiently increased and run on the most frequently updated large database of past records. The final individual obtained after satisfying the above mentioned convergence criteria is given in Table 4.

Table 4: database format of Final Individual

F	D1	D2
-202	-280	-321

The final individual thus obtained represents a product ID and excess or shortage stock levels at each of the two members providing essential information for supply chain inventory optimization.

5. Conclusion

Inventory management is an important component of supply chain management. We have proposed an innovative and efficient approach based on Particle Swarm optimization algorithm using MATLAB that is aimed at reducing the total supply chain cost by predicting the most probable surplus stock level and shortage level in each member of the supply chain for the forthcoming period.

REFERENCES

[1] Inventory Optimization in Supply Chain Management using Genetic Algorithm ,P.Radhakrishnan, V.M.Prasad and M. R. Gopalan, International Journal of Computer Science and Network Security, Volume No..9, Issue No.1, 2009.

[2] Multicriteria Inventory ABC Classification in an Automobile Rubber Components Manufacturing Industry, K.Balaji et al, Proceedings of the 47th CIRP Conference on Manufacturing Systems,2010.

[3] Cyclic versus reactive planning for inventory routing, Birger Raa, Procedia - Social and Behavioral Sciences, 2014.

[4]ATwo-Echelon Inventory Optimization Model with Demand Time Window Considerations, Wikrom Jaruphongsa et al., Journal of Global Optimization, Volume No.30, Issue No.1, 2004.

[5] Genetic Algorithm Based Multi Product and Multi Agent Inventory Optimization in Supply Chain Management, Tarun Kumar et al., International Journal of Modeling and Optimization, Volume No.2, Issue No. 6, 2012.

[6] Stochastic modeling of a two-echelon multiple sourcing supply chain system with genetic algorithm, Chingping Han Montri Damrongwongsiri, Journal of Manufacturing Technology Management, Volume No. 16, Issue No.1, 2005.

[7] Application of Genetic algorithm to supply chain inventory optimization, N.Jeyanthi et al, Journal of Contemporary Research in Management, January- March 2010

[8] Optimizing inventory and store results in big box retail environment, Amine Ayad, International Journal of Retail & Distribution Management, Volume No. 36, Issue No. 3, 2008.

[9] A genetic algorithm approach to the integrated inventory-distribution problem, Abdelmaguid T.F, Dessouky M.M, International Journal of Production Research, 44, Page: 4445-4464, 2006.

[10] Multi-matrix real-coded Genetic Algorithm for minimizing total costs in logistics chain network, Pongcharoen, P., Khadwilard, A. and Klakankhai, A., Proceedings of World Academy of Science, Engineering and Technology, Volume No. 26, 2007.

[11]Geometric Particle Swarm Optimization, Alberto Moraglio, Cecilia Di Chio,Julian Togelius and Riccardo Poli, Proceedings of Journal on Artificial Evolution and Applications, Volume No..2, 2008.

[12]Applying Particle Swarm Optimization to Schedule order picking Routes in a Distribution Center, Ling-Feng Hsieh, Chao-Jung Huang and Chien-Lin Huang, Proceedings of Asian Journal on Management and Humanity Sciences, Volume No.. 1, Issue No. 4, 2007.