Evolutionary Inventory Control for Multi-Echelon Systems

Dilay Çelebi

Abstract. The purpose of this chapter is to present the use of Genetic Algorithm (GA) for solving multi-echelon inventory problems. The literature of GA dealing with inventory control problems is briefly reviewed with particular focus on multi-echelon systems. A novel GA based solution algorithm is introduced for effective management of a stochastic inventory system across a distribution network under centralized control. To demonstrate the performance of proposed GA structure, several test cases with different operational parameters are studied and experimented. The percentage differences between the total cost obtained by GA and the lower bounds and simulation results are used as performance indicators. Findings of the experiments show that the proposed GA approach can be very useful for obtaining feasible and satisfying solutions for the centralized inventory distribution problem.

1 Introduction

Most consumer or industrial products are manufactured in and distributed through multi-echelon systems. Inventory control is critical in multi-echelon systems because of the financial necessity of maintaining a sufficient supply of products to meet both customers' needs and manufacturing requirements. Opportunity cost is the main component of inventory related costs; money tied up in inventories is not available for some other use. Inventories also create additional operational cost by consuming physical space, personnel time, and capital. Holding of inventories can cost anywhere between 20% and 40% of the product value, hence the effective management of inventory is critical in supply chain operations (Ballou, 1999).

The importance of a good inventory management in a supply chain is fully recognized by practitioners and researchers. Besides the traditional inventory management problems, the variability of orders increases in moving up from the

Dilay Çelebi
Istanbul Technical University, 34367 Maçka, Istanbul
e-mail: celebid@itu.edu.tr

M. Köppen et al. (Eds.): Intelligent Comput. Optimization in Eng., SCI 366, pp. 329–356.
springerlink.com

downstream members to upstream members in the supply chain. This phenomenon is known as *bullwhip effect* (Lee et al., 1997) which causes excessive inventory, loss of revenue, low customer service levels, and inaccurate production plans throughout supply chain systems. Much of the literature has shown that the bullwhip effect can be minimized through information sharing and synchronized inventory control in the supply chain (Cachon and Fisher, 2000). Hence the supply chain performance might be improved by the lowering of inventory levels and the reduction of the cycle times.

The scope of this chapter is confined to use of Genetic Algorithms (GA)s for handling operational issues of inventory control and management in multi-echelon inventory networks. The chapter is structured in six parts. Section 2 gives some fundamental definitions and briefly explains the complexity of multi-echelon inventory control problems. Section 3 provides a review of literature on use of GAs to solve multi echelon inventory control problems and evaluates the state of GA applications in these areas. The studies are classified into three categories according to the network structures they handle. Section 4 presents a novel GA structure for a stochastic lot sizing problem in a centralized distribution system. Model structure and the steps involved in development of the proposed GA scheme, such as chromosome representation, initialization, fitness function development, and determination of operational parameters are explained in detail in relevant subsections. The performance of GA is demonstrated through experiments conducted on several test cases with different operational parameters. Finally, section 5 discusses the issues and boundaries related to the application of the GA on inventory control problems. The chapter is closed with discussions for further research directions.

2 Inventory Control in Multi-Echelon Systems

A multi-echelon inventory system refers to a multistage production/inventory system in which each stage obtains its supply from its predecessor(s) and supplies its supply to its successor(s). Inventory control in multi-echelon systems deals with the problem of determining the best replenishment sizes of items at each stage, mostly with the purpose of minimizing the total cost of the system which usually covers the costs of carrying inventories, costs of making orders, costs of inter-location transfers, and costs of shortages.

Several multi-echelon inventory/production systems can be modeled as a serial system (Clark and Scarf, 1960). In a serial system, each installation has at most one predecessor and at most one successor. An illustration of a serial system is given in Figure 1. The customer demand only arises at the lowest level. Each installation is replenished from its predecessor and the highest installation replenishes from an outside supplier. It is, in general, considerably easier to deal with serial systems than with other types of multi-echelon systems. The main reason to discuss such systems is to obtain preliminary results to study more complex systems.

Fig. 1 A Serial Inventory System.

Within a manufacturing context, a final product is sometimes the result of a process which can be decomposed into several levels, broadly corresponding to assembly activities. As illustrated in Figure 2, in an assembly system, each installation has at most one immediate successor. In such systems, the safety stock levels should be positioned wisely and the assembly schedule should be done carefully for effectively managing the component procurement and maintaining service levels on the demand side.

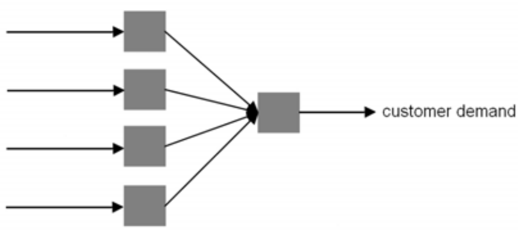

Fig. 2 An Assembly Inventory System.

Meanwhile, inventory distribution systems are generally divergent. A distribution system involves a number of installations at the lowest level which satisfy customer demand and in turn act as customers of higher level installations. Figure 3 shows such a system with two levels: a central warehouse and a number of retailers.

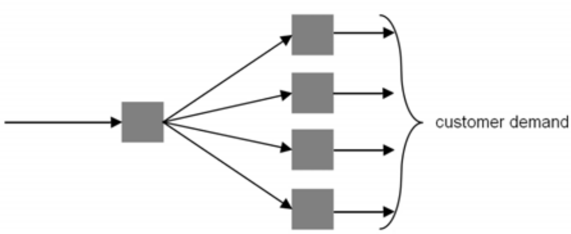

Fig. 3 A Distribution Inventory System.

There exists substantial amount of studies for multi-echelon inventory control, concerned with the analysis and modeling of systems under different operating parameters and modeling assumptions. Extended reviews about the topic might be found in van Houtum et al. (1996) and Gumus and Guneri (2007).

Two distinct configurations for multi-level inventory systems can be considered according to the center of management. First is the *decentralized* systems, where each member of the network takes replenishment decisions on its own and based on only local data. Though it is simple to construct and control such systems, it may not be the most effective. Recently, to increase the competitiveness and effectiveness of their supply chains, companies have begun to set cooperative agreements to manage inventory, which requires sharing demand information and setting mutually agreed upon performance targets for the supply chain. However, most of the time, it's mistakenly assumed that efficiency can be attained simply by sharing information and forming *strategic alliances* within supply chain partners (Silver et al., 1998). In fact, only few companies are able to fully exploit the advantages of collaboration in their supply chains (Holweg et al., 2005), because incorporating customer demand information into inventory control processes to develop sound inventory management is critical to long term survival and competitive advantage. It is important not only to exchange information, but equally, to alter the replenishment and planning decision structure so that a range of additional benefits can be achieved. As a result, a second type of systems become popular as *centralized* systems, in which the stock control activities of the whole system become concentrated within a particular member or group of members. These members take the full control of the inventory replenishment of the chain, and use demand and cost visibility in planning supply operations. The centralization of inventory management might provide cost reductions and improved service levels due to the decreased uncertainty and better utilization of resources for production and transportation (Waller et al., 1999).

Considering centralized solutions for inventory control in supply chains introduces computational difficulty. Schwarz (1973) shows that in a one-warehouse, multi-retailer situation, the form of the optimal policy can be very complex; in particular, it requires that the order quantity at one or more of the locations vary with time, even if all relevant demand and cost factors are time-invariant. Federgruen (1993) notes that algorithms for determining optimal strategies are complicated even for most deterministic demand systems, and complexity dramatically increases in models with stochastic demand.

For centralized control of multi-echelon systems, Clark and Scarf (1960) introduce the concept of *echelon stock*. Echelon stock consists of the stock at any given installation plus stock in transit to or on hand inventory at a lower installation. They have shown that order-up-to policies based on echelon stock inventories are optimal for serial inventory systems with periodic review. Their optimality results for serial systems are later generalized to an infinite time horizon by Federgruen and Zipkin (1984), to assembly systems by Rosling (1989), and to batch ordering by Chen (2000). However, determination of optimal lot sizes by provided models for large scale problems still suffers from computational burden. Moreover, it is not possible to show that echelon stock inventory policy is optimal for distribution systems due to the allocation problem, and for some cases a strategy based on echelon stock inventories might be inferior to a installation stock based strategy (Chen, 2003). The optimization of such systems requires the analysis of a multi-dimensional dynamic programming.

There isn't any known stochastic, multi-period, multi-location model capable of handling complex systems like inseparable cost structures and nonlinear transportation costs (Federgruen, 1993; Chen, 2003). In a practical setting, it is considered too difficult to solve the distribution lot sizing problem by dynamic programming numerically due to the curse of dimensionality. Even for the smallest number of retailers and periods, the exact solution is considered to be impractical (Federgruen and Zipkin, 1984).

3 GAs for Multi-Echelon Inventory Problem

During the last two decades, the opportunities for efficient control of multi-echelon inventory systems have increased substantially (Axsater, 2003). One reason is new information technologies which have created a completely different infrastructure and increased the possibilities for efficient supply chain coordination. Another reason is progress in research, which has resulted in new and efficient techniques for solving hard combinatorial optimization problems. Meta-heuristics such as tabu search, GA and simulated annealing, are examples of such tools which have become popular tools for solving multi-echelon inventory control problems due to computational complexity of such problems.

GA has received considerable attention regarding their potential as an effective optimization technique (Gen and Cheng, 2000). First pioneered by Holland (1975), GA is powerful stochastic search and optimization technique based on principles of natural selection and evolution that has been widely studied, experimented and applied in many fields in engineering (Goldberg, 1989; Holland, 1975). Many of the real world problems, which might be difficult to solve by traditional methods but are ideal for GA.

There exists wide range of studies which implement GA to cope with the multi-echelon inventory management problem. Table 1 provides a review of how GAs have been used to solve multi-echelon inventory problems and following sections give brief summaries of these studies, classified under three main categories according to the network structure they handle.

Table 1 Use of GAs in Multi-echelon Inventory Problems

Study	Objective	Network Structure	Demand Structure
Vergara et al. (2002)	Determination of product scheduling and replenishment sequences of products in a synchronized supply chain to minimize the total costs	Assembly	Deterministic
Syarif et al. (2002)	Minimization of the total transportation, inventory, order and warehouse costs while fully supplying deterministic demand	Distribution	Deterministic
Kimbrough et al. (2002) and O'Donnell et al. (2006)	Minimization of the bullwhip effect by in a serial supply chain model based on the MIT beer game	Serial	Stochastic
Yokoyama (2002)	Determination of the target order-up-to levels for Distribution Center (DC)s and transportation quantities to minimize the expected total costs in a single item distribution system	Distribution	Stochastic
Berretta and Rodrigues (2004)	Determination of the quantity to be produced in different periods in a planning horizon, such that an initially given demand forecast can be attained in a multistage production system with capacity constraints	Assembly	Deterministic
Daniel and Rajendran (2005)	Optimization of the inventory levels by a simulation-based GA to minimize the total supply chain cost	Serial	Stochastic
Han and Damrongwongsiri (2005)	Formulating a model to define stochastic, multi-period, two-echelon inventory with the many-to-many demand-supplier network problem to develop a (R,s) inventory management system	Distribution	Stochastic
Torabi et al. (2006)	Evaluation of optimal lot sizes and delivery schedule that would minimize the average of holding, setup, and transportation costs per unit time	Assembly	Deterministic
Wang and Wang (2008)	Implementation of (Han and Damrongwongsiri, 2005) to solve a real industry case	Distribution	Stochastic
Fakhrzad and Khademi Zare (2009)	Optimization of lot-sizes in a complex multi-stage production scheduling problems with production capacity constraint	Serial	Deterministic
Hnaien et al. (2009)	Optimization of a two-level assembly system under lead time uncertainties, where the finished product demand for a given due date is supposed to be known	Assembly	Stochastic

3.1 Serial Networks

Fakhrzad and Khademi Zare (2009) present a combination of GA with Lagrange multipliers for lot-size determination in a multi-stage, multi-product and multi-period production scheduling problem. First, the original problem is converted to several individual problems using a heuristic approach based on the limited resource Lagrange multiplier. Then, each individual problem has been solved using GA combined with one of the neighborhood search techniques. Each chromosome is represented by a $(T \times 2m)$ matrix where m is the number of elements and T is the number of periods. This representation consists of lot-size (X) and inventory values (I) for each element in each period which is illustrated as follows:

$$\begin{bmatrix} X_{11} & X_{12} & \dots & X_{1\mathrm{T}} & I_{11} & I_{12} & \dots & I_{1\mathrm{T}} \\ X_{21} & X_{22} & \dots & X_{2\mathrm{T}} & I_{21} & I_{22} & \dots & I_{2\mathrm{T}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_{\mathrm{m}1} & X_{\mathrm{m}2} & \dots & X_{\mathrm{mT}} & I_{\mathrm{m}1} & I_{\mathrm{m}2} & \dots & I_{\mathrm{mT}} \end{bmatrix}$$

Fitness function is developed to have two modes. One mode represents the cost value for the feasible solution and the other indicates the feasibility of the solution. A solution for the initial population is obtained using WagnerWhitin (WW) algorithm. Two different combinatorial operations, namely a memetic algorithm and WW combination, are used for crossover operation to generate new offsprings. Experiment results over a set of 60 test problems show that proposed hybrid algorithm solves the problem in much less time, with better solutions and lower costs compared to memetic algorithm and CPLEX solution.

Daniel and Rajendran (2005), study the performance of a single-product serial supply chain operating with a base-stock policy. A single period, multi-echelon, single product model is formulated to optimize the inventory (i.e. base stock) levels in the supply chain to minimize the total supply chain cost, comprising holding and shortage costs for all installations in the supply chain. A set of actual values of the base-stock levels are used to code all the genes in a chromosome that represent every member in the chain. The chromosome length is set equal to the number of installations, because every installation in the supply chain is assumed to operate with a particular base-stock level. Every chromosome in the population is evaluated through simulation and fitness value is computed by using the objective function value. Test results illustrate that the proposed GA performs superior to a random search procedures for defined experiment sets.

Both Kimbrough et al. (2002) and O'Donnell et al. (2006) manage to decrease the bullwhip effect by implementing GA on a serial supply chain model based on the MIT beer game. Each player in the game makes their own ordering decisions based on only the orders from the next downstream player. The inventory and backorder cost are calculated for every period in the simulation experiments with each unit cost being constant throughout and the sum of the two types of costs, called total cost, is used as the criteria for GAs to determine the optimal ordering policies.

3.2 Assembly Networks

Torabi et al. (2006) investigate the lot and delivery scheduling problem in a simple supply chain where a single supplier produces multiple components on a flexible flow line and delivers them directly to an assembly facility. The main objective is to find a lot and delivery schedule that would minimize the average of holding, setup, and transportation costs per unit time for the supply chain where the decision variables are production sequence vectors at each stage and machine. The problem is formulated as a mixed integer nonlinear program and a Hybrid Genetic Algorithm (HGA) is developed to solve the problem. The proposed HGA incorporates a neighborhood search into a basic GA that enables the algorithm to perform genetic search over the subspace of local optima.

Two different encoding schemes are considered to represent the discrete part of a solution for the problem. In the first format, each chromosome is composed of m sub-chromosomes, where m gives the number of stages. So that, each sub-chromosome represents the production schedule in that stage. At stages with only one machine, the corresponding sub-chromosomes are permutation vectors of size n where n is the number of components. At stages with multiple parallel machines, the corresponding sub-chromosomes are composed of a component symbol list and a partitioning symbol list, in which integers are used to represent sequence of components and asterisks are used to designate the partition of components to the machines. This first format is good for representing a complete solution for the problem, because it covers the entire solution space and there is a unique string associated with every solution for the problem. However, there are difficulties associated with crossover and mutation operations, so a new encoding scheme is considered. The second format relates to the set of permutation vectors of size n. Each permutation vector represents the order in which the given set of components is processed at each stage. Such a vector by itself does not specify the complete solution, so an appropriate procedure out of two constructive heuristics is used to construct a complete solution for every given permutation vector. The fitness function is set equal to the objective function of the problem and the optimal lot sizes associated to each solution representing a given production schedule is solved by a NLP.

Vergara et al. (2002) develop an evolutionary algorithm that calculates the production sequence at each supplier that would minimize transportation, setup, and inventory holding costs across a multi-component assembly system. The goal of the proposed GA is to determine a common delivery cycle time and production sequence of components for each member of a synchronized supply chain. Integer value representation is used for encoding the solutions. First three places in each chromosome contain the total cost, synchronized delivery time, and the minimal cycle time. The rest of the chromosome is composed of the production sequences of components for each supplier. The last space in the chromosome is used for holding the cost of the assembly center. The performance of the algorithm is tested through the comparisons with an enumeration procedure that identifies the global minimum. On the average, GAs are observed to find the global optimum in 97% of the cases and the error term is less than 0.0038 in the remaining cases.

Berretta and Rodrigues (2004) deal with the multistage capacitated lot-sizing problem with an objective of determining the production lot-sizes of multiple items that minimizes the production, inventory and setups costs subject to demand and capacity limitations. A memetic algorithm approach is used for solution of the problem. Each solution is represented by a matrix of size $2N \times T$ (where N is number of items and T number of periods), with lot-size and inventory of each item in each period. Each solution of the initial population is created using the WW algorithm. Fitness of each solution is composed of two values, one for value of objective function, the other for representing the feasibility of the solution. Each chromosome uses a local search algorithm to improve its current fitness value and transfers the best solution to the population.

Two different crossover schemes are used to stimulate diversity of solutions in the population. First crossover method determines a production lot size of offspring randomly over parent chromosomes and updates the inventory level accordingly. Second method uses WW algorithm, which changes the setup costs of some items randomly and develops a solution for each item by WW algorithm. The performance of the algorithm is tested through comparisons of solutions with lower bounds evaluated by Lagrangean Relaxation in three groups of instances. The average gaps between the lower bound and the heuristic solution are observed to be between than $11-12\%$.

Hnaien et al. (2009) examine supply planning for two-level assembly systems under lead time uncertainties. They handle an optimization problem for a single period system where the finished product demand for a given due date is supposed to be known. The objective is to find the component release dates in order to minimize the sum of the holding costs of the components and the shortage cost for the finished product. Each chromosome has been coded with an array of integer numbers where each gene of a chromosome represents an order release date. Therefore, with a chromosome length equal to number of components, a complete encoding that ensures that all solutions to the problem is represented and considered by the algorithm. The expected total cost of the system is used to evaluate the fitness of each individual of the population. Two selection phases are considered for generation of new generations. First is a reproduction selection to determine the individuals on which the evolutionary operators will be applied. Second is a replacement selection concerning the evolution of the population from a generation to the next. A standard single point crossover and mutation is then applied to the generated offsprings to introduce some diversity in the population. The mutuant gene is selected randomly at each generation by sampling a uniform random number. In addition to the typical genetic operators, also a local search is incorporated in order to speed up the convergence of the algorithm.

3.3 Distribution Networks

Syarif et al. (2002) consider a single period capacitated distribution problem which also consists of the binary decisions of opening plants and Distribution Center (DC).

The number of the DC's to be opened is assumed to be given as a constant. The objective is to minimize the total transportation, inventory, ordering and warehouse costs under capacity constraints, while fully supplying deterministic customer demand. A spanning tree based GA is used to solve the model. Vertex encoding is used with Prüfer number representation which establishes a unique sequence of length $n-2$ associated with the tree with n vertices. The chromosome consists of five sub-strings as illustrated in Figure 4. The first and the second substrings are binary digits representing opened/closed plants and DCs, respectively. The last three numbers are the Prüfer numbers to represent the production and distribution pattern for each echelon.

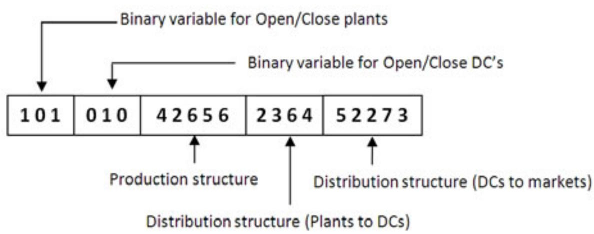

Fig. 4 Chromosome Structure Used by Syarif et al. (2002).

The infeasibility that may result from capacities and distribution structures are eliminated with a repair strategy, simply by replacing the digits in Prüfer numbers until the number of connections in the supplier set is equal to the number of connections in the supplier plants set for each node. A single point crossover operation is used. For mutation, an inversion-displacement operation is employed. Inversion selects two positions within a chromosome at random and inverts the substring between these two positions where displacement selects a substring at random and inserts it in a random position. The results of various experiments show that proposed algorithm provides near optimal solutions both for small and large scaled problems.

Yokoyama (2002) present a model and a solution procedure based on GA for single item distribution system with stationary and probabilistic demand. The objective is to determine the target order-up-to levels for DCs and transportation quantities to minimize the expected total inventory related costs and transportation costs. Simulation and linear programming are used for calculating the estimates of expected costs. Each solution is represented by a integer array of size equal to the number of the DC where each gene holds the order-up-to level for the given DC. Total expected cost for a given chromosome is estimated by simulation where optimal transportation quantities for given inventory levels are determined by linear programming. Two-point crossover operation, roulette wheel selection, and random replacement mutation operations are implemented for generating offsprings. The results of the algorithm are compared to the results of random local search algorithm. GA is observed to produce slightly better results than random search within same computation times.

Han and Damrongwongsiri (2005) develop a model to define stochastic, multi-period, two-echelon inventory with the many-to-many demand-supplier network problem to develop a (R,S) inventory management system where R refers to the replenishment period and S is the order-up-to level. GA is applied to derive optimal solutions through a two stage optimization problem. First stage covers the optimization of inventory order-up-to levels of the warehouses based on historical demand. Binary representation of the order levels is used for encoding where the length of the bitstream assigned to each warehouse is determined by the capacity of the warehouse. The fitness function is calculated as the sum of inventory carrying and shortage costs. The optimal inventory order-up-to levels determined in the first stage are used as inputs of the second stage, distribution planning. The goal of this stage is to determine the optimal transportation quantities of the retailers that minimizes inventory related costs. Binary encoding is preferred for chromosome representation, where the bit length assigned for each retailer is determined by both the capacity of the retailer and the total warehouses' maximum inventory level. First population is randomly generated and roulette wheel approach is used for selection. Crossover is done by one-cut-point method and random point mutation is used. Results of numerical experiments do not contain any performance comparisons with other methods but various experiment sets are presented to illustrate the flexibility of the method to handle many uncertainty factors.

The approach developed by Han and Damrongwongsiri (2005) is implemented to solve a real industry case by Wang and Wang (2008). Both the mathematical model and the GA structure are adopted for optimizing the distribution operations of a medical products manufacturer which supplies to four Nordic region markets from three geographically distinct warehouses. The demands at each market are assumed to be normally distributed with parameters forecasted through past data and independent of each other. Again no data is provided for performance comparisons but it's noted that GAs are able to compute the trade-off of all parameters and derive a good inventory and distribution plan, which might lead to a reduction up to 80% on the total cost of the system.

4 A GA Approach for Stochastic Lot-Sizing in a Centralized Distribution Network

This section presents a novel GA structure and the implementation issues for solution of a stochastic lot sizing problem in a centralized distribution system. To author's knowledge, this is the first study that investigates the use of GA for the capacitated lot sizing problem of One Warehouse - Multi Retailer (OWMR) system under stochastic and time varying demand. A similar study is given by Celebi and Bayraktar (2008) for deterministic demand case. The main contribution of the proposed GA structure is the domain specific encoding scheme and the fitness value calculation technique that uses dynamic programming for evaluating best order structure of the warehouse. This structure can be utilized as a collaborative supply chain planning tool to effectively manage the distribution process.

This section also presents an extensive numerical study over simulations which identifies the parameter settings where the proposed GA based method performs better or poorer than a widely used approximation technique. These results also implicitly shows the impact of cost parameters on the performance of the studied distribution network.

4.1 Model Description

A two-echelon distribution inventory system with a single central warehouse and multiple retailers is considered. The network is controlled by a central distributor which is occupied with all relevant information. That means, the distributor monitors the end customer demand and retailers' inventory levels in order to decide on order quantities, shipping and timing of replenishment orders.

Retailers directly replenish their stocks from the warehouse where warehouse orders from an outside supplier. It is assumed that the outside supplier have infinite source of supply or work at very high service levels so delays from the supplier side are negligible. All facilities follow a periodic inventory order policy where the lengths of planning periods are the same for all retailers and the warehouse. Customer demands are probabilistic and only placed in retail locations. It is assumed that the demand rates might change from one period to another, but remain constant within a period. This is not a restrictive assumption when the period length is kept small enough compared to the planning horizon. Moreover, this assumption is a good representative of the practical situation when demand quantities are forecasted by a time series method. In such a case there exists a demand forecast for given period, and the variations of demand within the period are estimated by a probability distribution.

An inventory problem for T periods is considered. There is a fixed ordering cost incurred with each replenishment with a cost function $\delta(q)$;

$$\delta(q_t^n) = \begin{cases} cq_t^n + K_t^n, & \text{if } q_t^n > 0 \\ 0, & \text{if } q_t^n = 0, \end{cases} \tag{1}$$

where K_t^n is the setup cost, c is the unit purchasing cost, and q_t^n is the replenishment quantity at period t. During each period, the stock on hand is decreased by an amount equal to the demand.

In addition to ordering cost, for all locations, carrying inventories incurs holding costs at a rate of h which is charged on the inventory level at the end of each period. Unfilled customer demands are fully back-ordered at retailer level and the retailer shortages are penalized a rate of π, the back-order cost per unit.

$$L(x_t^n) = \begin{cases} h\sum_{u=0}^{x_t^n}(x_t^n - u)P(u) + \pi\sum_{u=x_t^n}^{\infty}(u - x_t^n)P(u), & \text{if } x_t^n > 0 \\ \pi\sum_{u=0}^{\infty}(u - x_t^n)P(u), & \text{if } x_t^n \leq 0. \end{cases} \quad (2)$$

In equation (2), x_t^n is used to express the inventory level and $P(u)$ is the probability of observing u units of demand.

The allocation decision made for shipping to a total number of N retailers has direct impact on warehouse's costs. At the beginning of each period, warehouse allocates a total of $\sum_N q_t^n$ units to the retailers. Delivery of a replenishment arrives ℓ periods after the allocation decision, when the stock on hand is increased by the amount of the replenishment. Since customer transactions only occur at retailer points and the warehouse directly replenishes from an infinite supply source, there aren't any costs associated with inventory shortages on warehouse side. Inventory carrying costs for the warehouse for holding y units of inventory, is denoted by $H(y, q_t)$ and given as:

$$H(y, q_t) = h_0(y - \sum_N q_t^n) \quad (3)$$

In a centralized system the optimal policy is not necessarily the aggregation of individual optimal policies because of the dependencies among members and costs associated with those dependencies. The purpose of our model is to obtain the minimum total cost for the overall system which is formulated as follows:

$$\text{Minimize} \quad \sum_{t=1}^{T}\Big(H(y_t, q_t) + \delta(p_t) + \sum_{n=1}^{N}\big(L(x_t^n) + \delta(q_t^n)\big)\Big). \quad (4)$$

subject to

$$x_{t+1}^n = x_t^n - D_t^n + q_{t-\ell}^n \qquad n = 1, \dots, N \quad (5)$$

$$\sum_{n=1}^{N} q_t^n \leq y_t \qquad t = 1, \dots, T \quad (6)$$

$$x_t^n \leq C_n \qquad t = 1, \dots, T, n = 1, \dots, N \quad (7)$$

$$y_t \leq C_0 \qquad t = 1, \dots, T \quad (8)$$

The solution of model presented above for the lot sizing problem of the centralized system gives the minimum value of objective function (4) under given constraints. The first two terms in the objective function refer to total expected warehouse inventory costs. Last two terms are the sum of expected inventory related costs of all retailers. First constraint is a balance equation which adjusts the inventory levels between two consecutive periods. This is not a simple linear equation due to the stochastic variable D_t^n. Second constraint limits the number of shipped products from warehouse to all retailers with warehouse's on-hand stock in period t. Constraints (7) and (8) ensures that retailers' and warehouse's inventory holding capacities are not exceeded.

Due to the stock allocation problem of distribution systems, the optimality formulations are functions of distributor's and N retailers' inventory levels and can not be broken down in the form of independent formulations (Clark and Scarf, 1960). The state of the system at the beginning of period t can be described by the vector $(x_t, q_{t-\ell_n+1}, \ldots, q_{t-1})$. For any period $t \in 1 \ldots T$, the minimum total expected cost function is defined as $f_t(x_t, q_{t-\ell+1}, \ldots, q_{t-1}, y_t, p_{t-\ell_0+1}, \ldots, p_{t-1})$. Here x_t and q_t refers to the vector of the inventory levels and replenishment quantities of all retailers, y_t and p_t refers to the inventory level and replenishment quantity of the warehouse for period t. The optimal lot sizing policy in centralized system is given by one combined formulation due to the dependencies between retailers' and distributor's orders. The recursive formulation for period t then becomes:

$$\begin{aligned} & f_t(x_t, q_{t-\ell}, \ldots, q_{t-1}, y_t, p_{t-\ell_0}, \ldots, p_{t-1}) = \\ & \min_{0 \leq y_t \leq C_0} \Big\{ \min_{0 \leq \sum q_t \leq min(y_t, C_n)} \{ \delta(q_t) + \delta(p_t) + H(y_t, p_t) + L(x_t) \\ & + f_{t+1}(x_t + q_{t-\ell} - u, q_{t-\ell+1}, \ldots, q_t, y_t + p_{t-\ell_0} - \sum q_t, p_{t-\ell_0+1}, \ldots, p_t) P(u) \} \Big\}. \end{aligned} \tag{9}$$

Solving (9) recursively for T periods gives us optimal policy for allocation and inventory replenishment strategy of the overall system.

4.2 Motivation for Using Genetic Algorithms

For finding the global optimum convexity plays a crucial role in minimization problems. Finding a local optimum solution is an important step of solving the global problem, however it is not sufficient most of the time. Traditional optimization technique works by obtaining the zeros of a function's derivative, and testing for optimality. Such derivative tests obtain local information, and hence yield solutions that are locally optimal. Removing the convexity assumption on the function to be optimized, this method may prove severely inefficient, as it cannot provide anything more than local information. In such a case, the characteristics of the solution space should be investigated for ensuring the global optimum.

The problem presented in section 4.1 is proven to be NP-complete (Florian et al., 1980) and there is no known method to decompose the model into smaller ones. Besides the non convex behavior of total cost function, (4), distribution network lot sizing problem consists of variables that are diverse in their behavior, boundaries, and the probability distribution type. Moreover, the individual objective functions of the parties are not linear and multiple objectives can not be combined into a single metric. The combinatorial and sequential behavior of the two-echelon lot sizing problem can not be easily handled by traditional optimization techniques. Then it is reasonable to investigate a search algorithm to approximate the global minimum of the inventory distribution problem.

The known methods developed so far need considerable computational effort to obtain an optimal solution and so are only able to solve relatively small problems within a reasonable time. The problem requires a considerable computational burden when the problem instance is large. Even for a simple example which represents a very small instance of a problem, the total cost function might have multiple local minima (Çelebi, 2008). A search algorithm that only uses the gradient ascent will be trapped in a local optimum, but any search strategy that analyzes a wider area will be able to cross the local optimum and achieve better results.

GAs are capable of handling non-linear functions and can also deal with multiple objectives. They do not have any restrictions on the nature of data or mathematical requirements about the problem structure, unlike most traditional approaches. Due to their evolutionary nature, they can handle any kind of objective functions and constraints (linear or nonlinear) defined on discrete or continuous, or mixed search spaces (Gen and Cheng, 1997).

One of the difficulties when dealing with non convex optimization problems by search algorithms is that one often falls into local optima. When this happens, often the global optimum is then impossible to reach. The probabilistic evolution of operators makes GA very effective at performing global search and reaching global optima. The GA based solution methods have the advantage of being able to generate both convex and non convex points of the optimization curve, accommodate nonlinearities in the objective functions, and not be restricted by the peculiarities of a weighted objective functions (Scott et al., 1995).

One should keep in mind that as it is common with all heuristic methods, GAs cannot guarantee to locate the global optimum in a problem space in a finite time. But still, for some engineering problems such as many design and simulation tasks, the most desirable solution may not be the conventional global optimum but instead a solution representing a robust answer to the problem in hand is sought.

4.3 The Proposed GA Structure

The construction of a GA for any problem can be separated into four distinct and yet related tasks (Hou et al., 1994):

1. The choice of the representation of the solutions,
2. The determination of the fitness function,
3. The design of the genetic operators to be used for creation of new generations, and
4. The determination of the probabilities controlling the genetic operators.

Each of the above four components greatly affects the solution obtained as well as the performance of the GA. The summary of these steps involved in the proposed GA structure are described in detail in the following sections.

4.3.1 Encoding and Initialization

The most critical problem in applying a GA is in finding a suitable encoding of the examples in the problem domain to a chromosome. A good choice of representation will make the search easy by limiting the search space, a poor choice will result in a large search space. Our candidate solutions are combinations of all possible order quantities of each retailer and the distributor, for a number of T periods, hence the phenotype space P is the set of all such combinations. To design a GA defined by a representation of phenotypes from P, integer value representation is used where each chromosome represents retailers order up to levels for each period. Each chromosome takes the following sequence in the proposed encoding scheme:

$$\mathscr{C} = q_1^1 q_2^1 \dots q_T^1 q_1^2 \dots q_T^2 \dots q_1^N \dots q_T^N$$

Here, q_t^n is the direct value representation of the replenishment quantity for retailer n in period t. Each chromosome is a string of $N \times T$ genes, where N is the total number of retailers and T is the number of periods. That means, i^{th} gene in the sequence is the replenishment quantity for retailer $\lceil i/T \rceil$ in period $i(\text{mod } T)$. Each gene can take values between 0 and C_n which corresponds to the inventory carrying capacity of retailer n.

This design guarantees the completeness and the correctness requirements of encoding. Completeness is simply a consequence of using allocation quantities for *all retailers* in *all periods*. The correctness condition is provided by a simple check before fitness calculation. If the inventory on hand exceeds the capacity level for the given allocation quantity in any period, allocation quantity is updated to provide a feasible inventory level. Hence, feasibility of the chromosomes are kept in the legal domain without use of any additional constraint. First population is created with randomly generated individuals.

4.3.2 Fitness Evaluation and Selection

The role of the fitness function is to represent the requirements for improvement (Eiben and Smith, 2007) for a given individual. The quality of the given solution f, represented by a chromosome $\mathscr{C}$, is determined by the minimum expected total cost of the system given by equation (4). The total cost represented by any chromosome is evaluated in three steps:

1. Each retailer's expected costs for the replenishment scheme proposed by the chromosome are calculated by using equation (10) for total of T periods.

$$\begin{aligned} g_t^n(x_t^n, q_{t-\ell}^n, \dots, q_{t-1}^n) = \{ & \delta q_t^n + L(x_t^n) \\ & + \sum_{u=0}^{\infty} g_{t+1}(x_t^n + q_{t-\ell}^n - u, q_{t-\ell+1}^n, \dots, q_t^n) P(u) \}. \end{aligned} \tag{10}$$

2. The replenishment quantities are summed up to develop the aggregate allocation quantities ($\sum q_t$) and the ordering policy of the distributor is evaluated by dynamic programming formulation as given in equation (11).

$$\begin{aligned} f_t(y_t, p_{t-\ell_0}, \dots, p_{t-1}) = \min_{p_t \geq 0} \{ & \delta p_t + H(y_t, \textstyle\sum q_t) \\ & + f_{t+1}(y_t + p_{t-\ell_0} - \textstyle\sum q_t, p_{t-\ell_0+1}, \dots, p_t) \}. \end{aligned} \tag{11}$$

3. The objective function value is taken as the sum of the cost of warehouse (Eqn. (10)) and total cost of retailers (Eqn. (11)).

Resulting raw fitness scores are converted to values in a range that is suitable for the selection function. To avoid the effect of the spread of the raw scores, *Rank* fitness scaling method is used which scales the raw scores based on the rank of each individual instead of its score (Gen and Cheng, 1997). The selection of mating parents is done through *roulette wheel algorithm*. The pseudocode of this algorithm can be found in (Eiben and Smith, 2007).

4.3.3 Creation of New Generations

At each iteration, the current population is used to create the offsprings that make up the next generation through a *general replacement scheme*, so that, the chromosomes in the current population are completely replaced by the offspring. That means, population size is kept constant in its initial level through generations. The creation of next generation is conducted by three types of children. Figure 5 presents the schematic illustration of the three types of children.

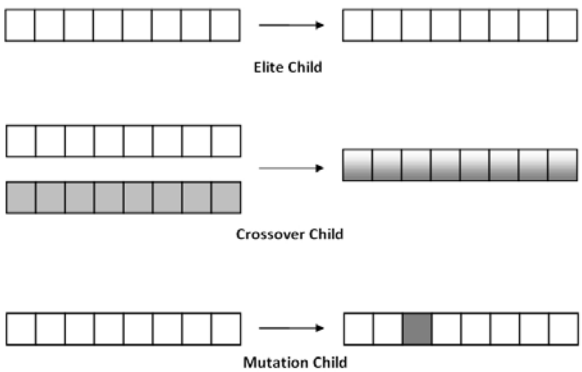

Fig. 5 Schematic Illustration of Three Types of Children.

Elite children are the individuals in the current generation with the best fitness values. These individuals are automatically passed to the next generation without any modification. Such an elitist algorithm is recorded to be able to speed up the performance of the GA significantly by preventing loss of good solutions once they are found (Zitzler et al., 1998). Experimenting on varying numbers (from 0 to 10) of elite chromosomes, the number of elite chromosomes is set to 2 which has given highest scoring individuals and provided best results in means of fitness value and time of convergence.

Crossover children are created by paring up the chromosomes and combining the vectors of a pair of parents. *Intermediate recombination*, which creates a new value for each gene of the offspring that lies between those parents. The function creates the child, c from parent$_1$, and parent$_2$ using the following formula:

$$\text{offspring} = \alpha \text{parent}_1 + (1-\alpha)\text{parent}_2,$$

where α is a random number generated from the range [0,1].

This method protects the feasibility of the chromosomes but might assign fractional numbers to the offspring genes. Such genes are repaired by rounding the number to the nearest integer for representing a valid order quantity.

Mutation children are created by introducing random changes, or mutations, to a single parent. To introduce variations into the chromosomes, *Random Resetting* in multiple points is implemented. Genes are selected according to a probability of being mutated, P_{mut}, which is defined by the *mutation rate*. Selected genes are then replaced with a value which is a realization of uniformly distributed random variable within the capacity range.

4.3.4 Setting Operational Parameters for GA Cycles

The selection of the best genetic parameters such as population size, number of generations, probability of crossover, and probability of mutation, is one of the important issues for the successful application of GA. Identifying the best parameters for a specific task is an open and challenging problem. Larger population sizes reduce the chance that GA will return a local minimum by searching the solution space more thoroughly but it also causes the algorithm to run more slowly. Experimenting on different population sizes, for the given problem instance it is observed that a population size of 50 gives satisfactory results both in sense of convergence speed and fitness values for our problem.

Two genetic operators, crossover and mutation, competes over the field of convergence. High crossover rate decreases the level of variation in the population so forces the convergence, while mutation forces diversity in the population. As a result of this fact, an optimum setting for the operator probabilities should have been determined. Optimal rates of these operators are problem specific and there are no defined rules on selecting the best GA operator fractions. To overcome this, the crossover and mutation rates are determined through several GA experiments for different rates of crossover and mutation by linear variations as suggested by Davis (1991). The experiment sets are composed by all combinations of 11 different crossover fractions over $[0.5, 1.0]$ and 11 different mutation rates over $[0, 0.2]$. The performance of each configuration is calculated by the value of the objective function, which is total system cost.

First one problem instance with 10 periods which represents the maximum period is used in our test cases. GA is run $121(11 \times 11)$ times to observe each configuration of crossover and mutation rate combinations. Two terminating conditions are set: First on the maximum number of generations, as 500, and second, on the number of generations without any improvement on fitness function, as 100. The minimum, average fitness function values of these three runs are recorded and crossover and mutation parameters are set to ones which give the minimum fitness values. The experiment results are presented in Figure 6.

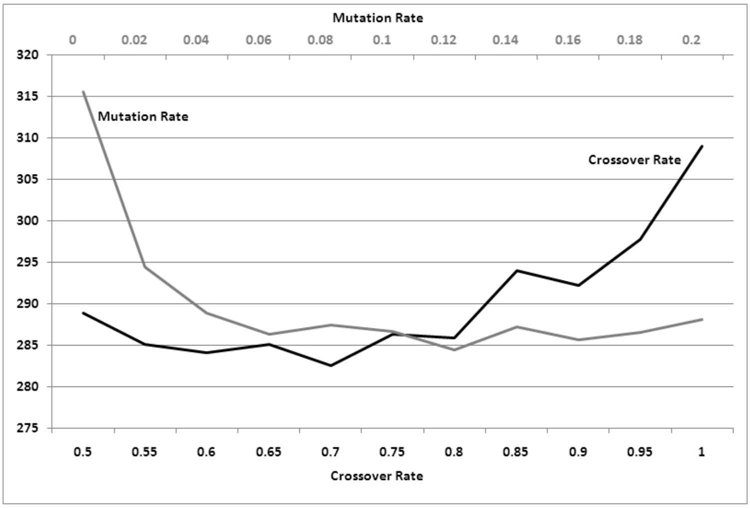

Fig. 6 Determination of Best Crossover and Mutation Rates.

Experiment results show that one of the crossover or mutation rates perform superior than others. Generally, high crossover rates are observed to give better results when mutation rate is also increased, however best results are obtained by a moderate crossover fraction, 0.7. On the average, mutation rate of 0.12 is observed to perform slightly better than others.

4.4 Numerical Study and Discussions

To demonstrate the performance of proposed GA approach, several test cases with different operational parameters are experimented. All algorithms are implemented in Matlab due to its efficiency for numerical computations, advanced data analysis capabilities, visualisation tools, and special purpose application domain toolboxes. The built-in functions of population creation, crossover, mutation, and fitness evaluation of Matlab-GA toolbox are modified according to the structure of the proposed GA design. Cost function and dynamic programming algorithms are coded as a common set of interdependent functions which are both used by GA algorithm and Balance Assumption (BA).

Since the optimal policy and the associated cost are unknown, instead of comparing the cost obtained by GA to the optimal cost, the cost of the system evaluated under BA and the cost realizations of system simulations(sim) are taken as the benchmark values. BA implies that in each period the downstream stock levels are balanced in such a way that a cost minimizing allocation without restrictions on the allocation variables will never result in negative allocation quantities. Thus, the system-wide cost calculated analytically under BA provides a lower bound for the true optimal cost. Detailed explanation of the balance assumption may be found in Eppen and Schrage (1981).

Assumption of zero lateral transshipment among retailers leads to a relaxation of the original optimization problem and is infeasible in real life. To provide a realistic benchmark value, an estimate for the real cost of the given policy (by BA) is obtained by simulation, so the cost of a feasible policy can be achieved. Each simulation is run for at least 100 times and terminated as soon as the width of a 99% confidence interval about the average cost function was within 1% of the average cost. The relative gaps between the results of the GA runs and the lower bound and the simulation runs ($\%\varepsilon_{lb} = 100 \times \frac{GA-LB}{LB}$ and $\%\varepsilon_{sim} = 100 \times \frac{SIM-GEN}{LB}$ respectively) are used as measures to assess the performance of the proposed GA method. Since the optimal cost of the original problem is between LB and GA, a small relative gap (ε_{lb}) implies that GA value is close to the optimal cost of the original problem, meaning that GA leads to an accurate approximation of the true optimal cost. On the other hand, even though the "balance assumption" might seem somewhat unrealistic, it has since been used extensively in the inventory literature and has been shown to produce solutions of very good quality in many different situations, (see for example Eppen and Schrage (1981); Federgruen and Zipkin (1984); van Houtum et al. (1996)). Policies that can provide considerable improvements over the BA in less or equal computation times, might be considered as well performing. Hence, a large relative gap between the simulation of "balance" policy and GA policy ($\%\varepsilon_{sim}$) is an indicator of the success of proposed GA structure for solving the given inventory distribution problem.

Due to the curse of dimensionality, only the case with two retailers with demands distributed over integers in $[0,4]$ with probabilities $[0.1, 0.2, 0.4, 0.3]$ is considered. This approximately corresponds to a moderate level of coefficient of variation. Both lead times for the retailers and the distributor are taken as 1. A limited number of test cases are structured by varying following cost parameters:

Fixed Costs: A variety of cases is considered for fixed replenishment costs defined by three different values for the retailers, $(K_n = 0, 5, 10)$ and three for the distributor:$(K_0 = 20, 10, 0)$).

Inventory Carrying Costs: Inventory carrying cost of retailers is taken constant, $h_n = 1$, and the variation is provided by changing the added value of the distributor: $(h_0 = 0.1, 0.5, 0.9)$.

Shortage Costs: The values of shortage costs are chosen as 4, 9, 19 and 99 which approximately correspond to no-stockout probabilities of 80%, 90%, 95% and 99% respectively.

A full factorial design is used to generate experimental cases that corresponds to 108 problem instances. All test cases are set for 10 periods.

First the lower bound and relevant simulation values for each test case are calculated. Then GA is run three times for each test and the run is stopped when either a pre-specified number of searches reaches to 1000, or there is no improvement in the best fitness value for 100 generations. The performance of each configuration is calculated by the value of the objective function, which is the expected total system cost. The best of these three runs which provides the minimum cost is used. The relative gap measures ($\%\varepsilon_{lb}$ and $\%\varepsilon_{sim}$) for scenarios $1-108$ are graphically depicted in Figure 7.

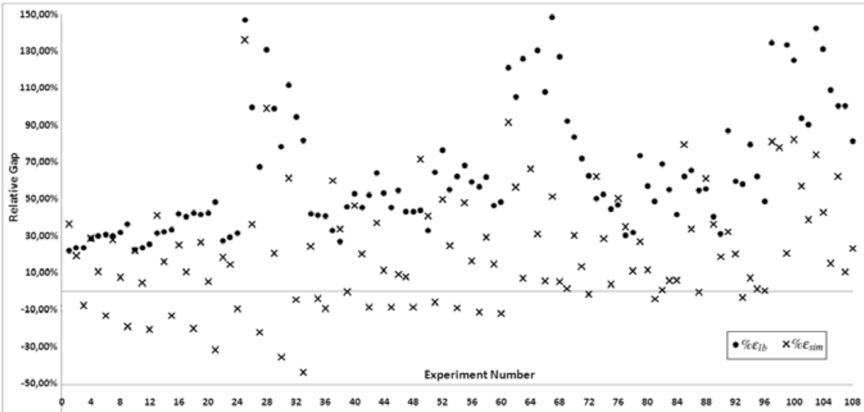

Fig. 7 Experiment Results.

When these 108 test instances are ranked with respect to $\%\varepsilon_{lb}$, 61 of them have $\%\varepsilon_{lb} > 50\%$. Among these cases only 11 of them have a $\%\varepsilon_{sim} < 0$, that means for remaining 50 cases, balance assumption fails to provide a good lower bound and an effective replenishment policy. In general, GA produced better results than balance assumption in 82 out of 108 cases with an $\%\varepsilon_{sim} > 0$. Savings over the total cost that are achieved by use of GA is 21.82% on the average, while savings up to 136.23% are recorded.

In order to see the influence of the cost parameters on the results, test results are also given in Tables 2 and 3, where data is summarized with respect to one parameter at a time. For example, the left part of Table 2 is dedicated to display the effect of the fixed replenishment costs of retailers. The first column gives the values of various measures for a set of 36 test instances in which $K_n = 0$. The measures used in the analysis are minimum, maximum and average $\%\varepsilon_{lb}$ (denoted by $\%\varepsilon_{lb}^{-}$, $\%\varepsilon_{lb}^{+}$ and $\%\bar{\varepsilon}_{lb}$, respectively), the minimum, maximum and percentage of improvement (denoted by $\%\varepsilon_{sim}^{-}$, $\%\varepsilon_{sim}^{+}$ and $\%\bar{\varepsilon}_{sim}$, respectively), and the number of the cases that GA produced better and worse results than the simulation of balance assumption (denoted by $\oplus$ and $\ominus$, respectively).

The findings can be summarized as follows:

1. In the test bed of 36 problem instances with $K_n = 0$, there are 14 scenarios with $\%\varepsilon_{sim} < 0$. This number decreases with increasing value of K_n. Similarly, the average improvement is 13.31% when retailer fixed replenishment costs are zero, and the improvement increases up to 30.79% with increasing value of K_n. This is in line with expectations. Balance assumption implies zero transshipment costs

Table 2 The summary of the results - Fixed Replenishment Costs

	Retailers' Fixed Costs			Distributor's Fixed Costs		
	$K_n = 0$	$K_n = 5$	$K_n = 10$	$K_0 = 0$	$K_0 = 10$	$K_0 = 20$
$\%\varepsilon_{lb}^{-}$	22.19	26.98	30.37	40.69	27.22	22.19
$\%\varepsilon_{lb}^{+}$	147.06	173.80	165.65	173.80	86.96	73.57
$\%\bar{\varepsilon}_{lb}$	48.83	71.81	75.37	105.35	50.56	41.32
$\%\varepsilon_{sim}^{-}$	−22.98	−12.13	−4.13	−44.08	−31.86	−20.79
$\%\varepsilon_{sim}^{+}$	136.23	91.69	82.28	136.23	79.45	62.23
$\%\bar{\varepsilon}_{sim}$	13.31	22.59	30.79	33.34	17.35	14.78
$\oplus$	22	27	33	29	26	27
$\ominus$	14	9	3	7	10	9

Table 3 The summary of the results - Inventory Carrying and Penalty Costs

	Inventory Carrying Costs			Penalty Costs			
	$h_0 = 0.9$	$h_0 = 0.5$	$h_0 = 0.1$	$\pi = 4$	$\pi = 9$	$\pi = 19$	$\pi = 99$
$\%\varepsilon_{lb}^{-}$	22.19	23.40	23.68	22.19	28.22	29.85	22.49
$\%\varepsilon_{lb}^{+}$	173.80	165.65	133.42	165.65	173.80	148.52	100.47
$\%\bar{\varepsilon}_{lb}$	74.57	63.82	58.84	68.26	67.50	74.02	53.19
$\%\varepsilon_{sim}^{-}$	0.58	−4.73	−44.08	−22.19	−35.84	−44.08	−20.79
$\%\varepsilon_{sim}^{+}$	136.23	78.01	38.72	136.23	99.09	74.05	62.17
$\%\bar{\varepsilon}_{sim}$	48.80	21.42	−4.76	36.43	27.44	14.60	8.82
$\oplus$	36	34	12	21	22	19	20
$\ominus$	0	2	24	6	5	8	7

between retailers thus increasing transshipment costs decreases the effectiveness of the method. That means when retailer replenishment costs are high, GA may provide better solutions than the policies based on balance assumption.

2. Similarly, when the distributor's replenishment costs are high, GA tend to perform better on the basis of comparisons to LB. This is not parallel to comparisons to SIM. Though increasing K_0 doesn't have a visible impact on the number of cases that GA performs better, the level of average improvement decreases dramatically.
3. Highest impact on performance of GA is observed on added value of the transportation to retailers. The performance of GA is top when the costs of carrying inventories in the warehouse is high. When $h_0 = 0.9$, GA never performed poorer than the policy based on balance assumption even though there is not significant change on the value of $\%\bar{\varepsilon}_{lb}$. The average improvement on the cost value is almost 50% where it reaches up to 136.23%. On the other hand, GA perform poorest when the carrying costs of inventories in the warehouse is comparably lower than carrying costs of retailers. When $h_0 = 0.1$, policy based on balance assumption produces better results than GA in 67% of the cases.

4. The value of the penalty costs has the lowest impact of GA performance. No trend can be observed on the number of the cases GA performs better and the value of the $\%\bar{\varepsilon}_{lb}$ with increasing value of the penalty costs, but the average rate of improvement $\%\bar{\varepsilon}_{sim}$ decreases visibly with increasing value of the penalty costs. The reason is not that GA performs poorer in such cases but balance assumption based policy performs better. For example, when π increases from 4 to 99, the average improvement of GA drops from 36.4% to 8.82% but the average gap between the GA and the lower bound, $\%\bar{\varepsilon}_{lb}$, also decreases from 68.26% to 53.19%.

The experiment results show that GA generally produce better results than the policy based on balance assumption. However in some cases, the solution provided by GA give higher total system costs than the benchmark. This shows that as in common with all heuristic methods, GAs cannot guarantee to locate the global optimum in a problem space in a bounded time. This is mainly through stochastic search behavior of the GAs. The results can be improved by increasing the number of trials and the computation time of the algorithm, and also experimenting on different GA operators such as population size, mutation and crossover. Besides, in practice the most desirable solution may not be the conventional global optimum but instead a solution representing a robust answer to the problem in hand is sought. Hence, for a large system with a high number of periods and retailers, GAs can be used as an effective algorithm for solving the multi-echelon inventory distribution problem under stochastic demand.

This study only presents the comparison of the proposed GA method with most known and used heuristic, for a system under a limited number of parameters. The experiment results can be strengthened by a more comprehensive numerical study, specifically targeting to assess the performance of the proposed heuristic. Another extension can be comparison of these results with the results obtained by other heuristic solutions, as well as the real optimal solution of the system.

5 Conclusion

GAs have been applied to a wide variety of multi-echelon inventory control problems in various studies. Tests on artificial data sets show that GA are pretty successful for determining a good solution even for the most complicated problems. However, some barriers might exist for the successful implementation of the proposed methods to real life. The complexity of today's business world means that it is often not possible to link external sources of information into the vendor's production and inventory control processes (Stank et al., 2001), as in many cases the same level of detailed information cannot be obtained from all of the distribution channels. For some environments, centralization may be expensive, very complex, or the coordination may be too much of a burden. This is especially true for large systems, which would require substantial computational power to store and process large amounts of information for centralized decision-making. A practical contribution can be made if GAs are applied to industrial inventory problems as integrated with interactive decision support systems where application data and test results on the algorithms performance are collected from real life applications.

Main assumption of multi-stage inventory control is the share of information among the supply chain members, however in practice, this assumption might be restrictive. The companies involved in strategic or incidental supply-network partnerships might be not willing or prepared to share information needed required for coordination of supply chain. Hence, the participants might agree to share only partial information due to the unwillingness of the participants to share private information such as cost structure. A distributed algorithm based on evolutionary algorithms that allows the distributed system to perform just as well as a centralized one may be designed for such cases. For example, Shin (2007) propose a framework for such a collaborative coordination mechanism: The coordinator solves the aggregate problem and delivers the solution results to all members. Each member evaluates the performance of the delivered solution from the coordinator using its own cost structure, solves its own problem in terms of its own objectives and measures the performance, calculates its penalty, and returns the penalty with its solution to the coordinator. Then, the coordinator selects the facility with the largest penalty value, modifies and solves the problem again, and redistributes the solution results to the all members. Not only the local optimization procedures but the collaboration mechanism can easily be optimized by GA to provide a global optimal solution.

For most of the multi-echelon systems, uncertainty is an unavoidable factor of inventory control and recognized to have a major impact on the manufacturing and service functions (Wilding, 1998). Uncertainties such as high variability in demand, manufacturing processes or supply create problems in planning, scheduling and control that jeopardize delivery performance (Fisher et al., 1997). Incorporating uncertainty might pose severe problems for the current GA structures developed for multistage inventory control. Explicitly, incorporating uncertainty will undoubtedly result in very complex models. However, the power of GAs to deal with such complex models proposes a promising topic of investigation and new research opportunities.

Nomenclature

ℓ Transportation lead time, page 13
π Unit Shortage Cost, page 13
c Unit purchasing cost, page 12
C_0 Maximum replenishment quantity for the distributor, page 14
C_n Inventory holding capacity of retailer n, page 14
D_t^n Demand faced by retailer n in period t, page 13
h Unit Inventory Holding Cost, page 12
K_t^n Fixed Cost per Order of retailer n at period t, page 12
$L(x)$ One period expected inventory carrying and shortage penalty costs, page 13
N The number of retailers, page 13
$P(u)$ Probability of observing u units of demand, page 13
P_{mut} Mutation Rate, page 18
q_t^n Replenishment quantity of retailer n at period t, page 12
T Number of periods in planning horizon, page 12
x_t^n Inventory level of retailer n in the beginning of period t, page 13

Acronyms

BA Balance Assumption
DC Distribution Center
GA Genetic Algorithm
HGA Hybrid Genetic Algorithm
OWMR One Warehouse - Multi Retailer
WW WagnerWhitin

References

Axsater, S.: Serial and distribution inventory systems. In: Kok, Graves (eds.) Handbooks in Operations Research and Management Science:V11 Supply Chain Management: Design, Coordination and Operation, pp. 525–559. Elsevier Ltd., Oxford (2003)

Ballou, R.H.: Business logistics management: planning, organizing, and controlling the supply chain. Prentice-Hall, Englewood Cliffs (1999) ISBN 0137956592

Berretta, R., Rodrigues, L.F.: A memetic algorithm for a multistage capacitated lot-sizing problem. International Journal of Production Economics 87, 67–81 (2004)

Cachon, G.P., Fisher, M.: Supply chain inventory management and the value of shared information. Management Science 46(8), 1032–1048 (2000)

Çelebi, D.: Stochastic Lot Sizing in a Centralized Distribution Network. PhD thesis, Istanbul Teknik Üniversitesi (2008)

Celebi, D., Bayraktar, D.: A genetic algorithm for multi location inventory problem. In: Proceedings of Fifteenth International Working Seminar on Production Economics (2008)

Chen, F.: Optimal policies for multi-echelon inventory problems with batch ordering. Operations Research 48(3), 375–389 (2000)

Chen, F.: Information sharing and supply chain coordination. In: Kok, Graves (eds.) Handbooks in Operations Research and Management Science:V11 Supply Chain Management: Design, Coordination and Operation, pp. 341–451. Elsevier Ltd., Oxford (2003)

Clark, A.J., Scarf, H.: Optimal policies for a multi-echelon inventory problem. Management Science 6(4), 475 (1960)

Daniel, J.S.R., Rajendran, C.: A simulation-based genetic algorithm for invertory optimization in a serial supply chain. International Transactions in Operational Research 12, 101–127 (2005)

Davis, L.: Handbook of Genetic Algorithms. van Nostrand Reinhold, New York (1991) ISBN 0-442-00173-8

Eiben, A.E., Smith, J.E.: Introduction to Evolutionary Computing. Natural Computing Series. Springer, Heidelberg (2007) ISBN 978354040184-1

Eppen, G., Schrage, L.: Centralized ordering policies in a multi-warehouse system with leadtimes and random demand. In: Schwarz, L. (ed.) Multi-Level Production/Inventory Control Systems: Theory and Practice, pp. 51–69, North Holland, Amsterdam (1981)

Fakhrzad, M.B., Khademi Zare, H.: Combination of genetic algorithm with lagrange multipliers for lot-size determination in multi-stage production scheduling problems. Expert Systems with Applications 36, 10180–10187 (2009)

Federgruen, A.: Centralized planning models. In: Graves, Kan, R., Zipkin, P. (eds.) Handbooks in Operations Research and Management Science:V4 Logistics of Production and Inventory, pp. 133–173. Elsevier Ltd., Oxford (1993)

Federgruen, A., Zipkin, P.: Approximations of dynamic, multilocation production and inventory problems. Management Science 30(1), 69 (1984)

Fisher, M.L., Hammond, J., Obermeyer, W., Raman, A.: Configuring a supply chain to reduce the cost of demand uncertainty. Production and Operations Management 6(3), 211–225 (1997)

Florian, M., Lenstra, J.K., Rinnooy Kan, A.H.G.: Deterministic production planning: Algorithms and complexity. Management Science 26, 669–679 (1980)

Gen, M., Cheng, R.: Genetic Algorithms and Engineering Design. John Wiley and Sons, New York (1997) ISBN 0-471-12741-8

Gen, M., Cheng, R.: Genetic Algorithms and Engineering Optimization. John Wiley and Sons, New York (2000 ISBN 0-471-31531-1

Goldberg, D.E.: Genetic Algorithms in Search, Optimization & Machine Learning. Addison-Wesley, Harlow (1989)

Gumus, A.T., Guneri, A.F.: Multi-echelon inventory management in supply chains with uncertain demand and lead times: literature review from an operational research perspective. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture 221(10), 1553–1570 (2007)

Han, C., Damrongwongsiri, M.: Stochastic modeling of a two-echelon multiple sourcing supply chain system with genetic algorithm. Journal of Manufacturing Technology Management 16(1), 87–107 (2005)

Hnaien, F., Delorme, X., Dolgui, A.: Genetic algorithm for supply planning in two-level assembly systems with random lead times. Engineering Applications of Artificial Intelligence 22, 906–915 (2009)

Holland, J.: Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor (1975)

Holweg, M., Disney, S., Holmstrom, J., Smaros, J.: Supply chain collaboration: Making sense of the strategy continuum. European Management Journal 23, 170–181 (2005)

Hou, E.S.H., Ansari, N., Ren, H.: A genetic algorithm for multiprocessor scheduling. IEEE Transactions on Parallel and Distributed Systems 5(2), 113–120 (1994)

Kimbrough, S.O., Wu, D.J., Zhong, F.: Computers play the beer game: Can artificial agents manage supply chains? Decision Support Systems 33, 323–333 (2002)

Lee, H.L., Padmanabhan, V., Whang, S.: Information distortion in a supply chain: The bullwhip effect. Management Science 43(4), 546–558 (1997)

O'Donnell, T., Maguire, L., McIvor, R., Humphreys, P.: Minimizing the bullwhip effect in a supply chain using genetic algorithms. International Journal of Production Research 44, 1523–1543 (2006)

Rosling, K.: Optimal inventory policies for assembly systems under random demands. Operations Research 37, 565–579 (1989)

Schwarz, L.B.: A simple continuous review deterministic one-warehouse n-retailer inventory problem. Management Science 19(5), 555–566 (1973)

Scott, E., Eheart, J.W., Ranjithan, S.: Using genetic algorithms to solve a multiobjective groundwater monitoring problem. Water Resources Research 31, 399–410 (1995)

Shin, H.J.: Collaborative production planning in a supply-chain network with partial information sharing. International Journal of Advances Manufacturing Technology 34, 981–987 (2007)

Silver, E.A., Pyke, D.F., Peterson, R.: Inventory Management and Production Planning and Scheduling, 3rd edn. John Wiley and Sons, West Sussex (1998) ISBN: 0471119474

Stank, P., Keller, S., Daugherty, P.: Supply chain collaboration and logistical service performance. Journal of Business Logistics 22(1), 29–49 (2001)

Syarif, A., Yun, Y., Gen, M.: Study on multi-stage logistic chain network: a spanning tre-based genetic algorithm approach. Computers & Industrial Engineering 43, 299–314 (2002)

Torabi, S.A., Fatemi Ghomi, S.M.T., Karimi, B.: A hybrid genetic algorithm for the finite horizon economic lot and delivery scheduling in supply chains. European Journal of Operational Research 173, 173–189 (2006)

van Houtum, G.J., Inderfurth, K., Zijm, W.H.M.: Materials coordination in stochastic multi-echelon systems. European Journal of Operational Research 95(1), 1–23 (1996)

Vergara, F.E., Khouja, M., Michalewicz, Z.: An evolutionary algorithm for optimizing material flow in supplpy chains. Computers & Industrial Engineering 43, 407–421 (2002)

Waller, M., Johnson, M.E., Davis, T.: Vendor managed inventory in the retail supply chain. Journal of Business Logistics 20(1), 183–203 (1999)

Wang, K., Wang, Y.: Applying genetic algorithms to optimize the cost of multiple sourcing supply chain systems an industry case study. Studies in Computational Intelligence 92, 355–372 (2008)

Wilding, R.: The supply chain complexity triangle: uncertainty generation in the supply chain. International Journal of Physical Distribution & Logistics Management 28(8), 599–616 (1998)

Yokoyama, M.: Integrated optimization of inventory distribution systems by random local seach and a genetic algorithm. Computers & Industrial Engineering 42, 172–188 (2002)

Zitzler, E., Deb, K., Thiele, L.: Comparison of multi-objective evolutionary algorithms: Empirical results. Evolutionary Computation 8(2), 173–195 (1998)