AN ESPC ALGORITHM BASED APPROACH TO SOLVE INVENTORY DEPLOYMENT PROBLEM

Felix T. S. Chan, V. Kumar and T. C. Wong,
Department of Industrial and Manufacturing Systems Engineering,
The University of Hong Kong, Pok Fu Lam Road, Hong Kong
E-mail: h9901225@hkusua.hku.hk, ftschan@hkucc.hku.hk, vk211@exeter.ac.uk

ABSTRACT
Global competitiveness has enforced the hefty industries to become more customized. To compete in the market they are targeting the customers who want exotic products, and faster and reliable deliveries. Industries are exploring the option of satisfying a portion of their demand by converting strategically placed products, this helps in increasing the variability of product produced by them in short lead time. In this paper, authors have proposed a new hybrid evolutionary algorithm named Endosymbiotic-Psychoclonal (ESPC) algorithm to determine the amount and type of product to stock as a semi product in inventory. In the proposed work the ability of previously proposed Psychoclonal algorithm to exploit the search space has been increased by making antibodies and antigen more cooperative interacting species. The efficacy of the proposed algorithm has been tested on randomly generated datasets and the results obtained, are compared with other evolutionary algorithms such as Genetic Algorithm (GA) and Simulated Annealing (SA). The comparison of ESPC with GA and SA proves the superiority of the proposed algorithm both in terms of quality of the solution obtained, and convergence time required to reach the optimal /near optimal value of the solution.

KEY WORDS
Lead Time, ESPC, GA, SA, Inventory

1. Introduction

In the present scenario manufacturing enterprises are in an environment where markets are frequently shifting, new technologies are continuously emerging, and contenders are multiplying globally. To adopt according to this situations industries are building their strategies in a way to support global competitiveness, new product advancement, and rapid market responsiveness. Rapid advances in technology and changes in demand patterns to incorporate customized features in manufactured products, and the relatively shorter life cycle has reallocated the emphasis of manufacturing strategy from mass production to small batch manufacturing, and has enforced the industries to adopt Make-to-Order (MTO) strategy. However, in order to reduce lead times, some proportion of production is planned in advance in accordance with the forecast of orders.

The changed market scenario has enforced the industries to enhanced competence. The technology to produce exotic grades and to customize finishing operations has positioned the modern industries to respond to more customized finished products. Reliable deliveries of the customized product are required to be synchronized with the customized production schedules. These aspects have put great pressure on industries to increase the product variety, and reduce the delivery lead time for a subset of their customers. Though, the necessity to have a large product variety and swift response time, places inconsistent demands on the production system. The reduction of delivery lead times has pressurized the industries to alter from pure Make-to-Order (MTO) production mode to a hybrid MTO/Make-to-Stock (MTS) mode. Preserving stocks of semi-finished products reduces the order accomplishment delay relative to the pure Make-to-order system. The semi-finished inventory is converted to finished product for the customers who agree to forfeit premium for it. The delivery through MTO mode, where production is not initiated until the customer order is received, continues for the other existing customers. As high degree of demand uncertainty exits in the market and the custom nature of the end product prevails, it is not economical to put the semi-finished inventory into stocks for all customers. The additional storage space requirement by the semi-finished inventory puts extra burden of holding cost on the industry, which is economically not viable. Modern Industries are facing the problem of finding an effective way to determine the position, and amount of strategic inventory to hold. Inventory deployment came for rescue to industries in this regard.

Industries are susceptible to uncertainty which is a major factor affecting its inventory planning. Prominent causes of uncertainties such as extended cycle times, volatile scenario of the markets, supply uncertainty, shortage or excessive production of orders not only affect the

planning but almost it damages the inventory planning. Due to tentative customer orders production, deviations from the actual orders lead to the shortage or extra inventory. To overcome this efficient planning strategy needs to be implemented. The present paper proposes an Endo-Symbiotic-Psychoclonal (ESPC) Algorithm to solve these existing complexities in efficient manner. To prove the efficacy of the proposed algorithm intensive runs have been carried out on computer simulated dataset and the results obtained are compared with the GA and SA solutions.

The paper is organized as follows. Section II deals with the literature review. Section III deals with the model formulation starting with the problem description, leading to IDP problem formulation. Section IV focuses on the background and steps of the Endo-symbiotic Psychoclonal (ESPC) algorithm. Section V explains the numerical experiment and Section VI concludes this paper.Clearly explain the nature of the problem, previous work, purpose, and contribution of the paper.

2. Literature Review

In this proposed work the problem of our interest is correlated to various dissimilar areas of research including the field of random yield, multi product substitutable inventory, and stochastic fixed charge network flow problems. Multi-product inventory problem with downward substitution and random demand was first studied by [1], in which the main emphasis was given on the conditions of the myopic ordering policy in a multi-period setting. [2] presented a mixed model for U-line balancing problem. The solution methodology for the downward substitution of a two-stage stochastic linear programming formulation (2S-SLP) of large-scale multi-product problems was considered by [3]. The model proposed in this paper incorporates various characteristics of replacement models but also oversimplifies to cases other than downward substitution. The paper takes account of the uncertainties such as due to Bullwhip effect in which the variability of the estimates or the forecasts of customer demand seems to amplify as the orders move up the supply chain from the customer, through retailers and wholesalers to the producer of the product or service [4]. The uncertainties affecting the inventory planning has been well established in the proposed article. One frequently suggested strategy for reducing the magnitude of the bullwhip effect is to centralize demand information, i.e., to make customer demand information available to every stage of the supply chain. Our model assumes stochastically proportional yield losses i.e. the number of good items in a lot is the product of a random yield rate with arbitrary distribution and the lot size and it is applicable to large-scale problems involving multiple products with substitution.
The multi-location inventory problem with transshipment between locations has been studied by [5], [6], and others. The work on Make-to-Stock (MTS) queues has been done by [7]. The Genetic Algorithm optimization of inventory control system by has been studied by [8]. The problems of two-stage stochastic linear programming have been solved by specialized computational procedures developed by [9]. The studies related to the modeling of the problem that determine the optimal point of differentiation, subject to a service-level constraint ([10], [11], [12], [13], [14], [15] etc.) have been incorporated in our model. The aims of the authors are to confine the advantages of inventory pooling when order-up-to-level inventory models are used to elicit replenishment. The model considered can be classified as a stochastic fixed-charge-network flow problem [16]. A dual-based procedure for the stochastic un-capacitated facility location problem has been studied [17]. [18] studied exact solution procedures for a location problem with stochastic demands in which facility capacities (inventory levels) are chosen a priori. [19] studied a multi-product inventory model with downward substitution and fixed setup costs. The considered model can be clearly distinguished from their work, as they assume perfect yield, no shortage constraints (rather, fixed setup costs), and take advantage of the downward substitution structure to propose simulation-based heuristics.

These shortcomings of previously applied algorithms on inventory deployment problem motivated the authors to develop a Meta heuristic, which is capable of escaping local optima. To enable this, the authors have extended previous approach of Psychoclonal ([20], [21]) by incorporating feature of reciprocal changes between the antibodies and antigens. The proposed algorithm termed as Endosymbiotic-Psychoclonal (ESPC) Algorithm, enjoys its flavor from Endosymbiotic algorithm [22] and Psychoclonal algorithm. The Endosymbiotic algorithm is based on the evolution process of eukaryotes from prokaryotes. An endosymbiont is an individual formed by the integration of two types of symbionts, in ESPC algorithm antigens (Ag's) an antibodies (Ab's) are modeled as symbionts, and the toroid matrix formed by the series of reciprocal changes refer as endosymbiont. Psychoclonal algorithm in ESPC algorithm inherits its attributes from Maslow's need hierarchy theory and the Artificial Immune System (AIS) approach. There are different levels of needs arranged in a hierarchy, namely physiological needs, safety needs, growth needs, esteem needs, and self-actualization needs. Clonal selection explains the response of immune systems when a non-self antigenic pattern is recognized by antibodies. Characteristic features of immune systems are immune

memory, hypermutation and receptor editing. The next section discusses the problem formulation in detailed.

3. Problem Formulation

A. Problem Description

The model considered has been taken from [15]. The model considered in this paper deals with the Inventory Deployment Problem (IDP). It considers basically two types of planning decisions: (i) strategic planning decisions made on an infrequent basis (e.g. quarterly or biannually); and (ii) operational planning decisions made more frequently (e.g., weekly or monthly). IDP refers to the problem of determining the demand volume, customer's priorities and type of orders to be served in the Make-To-Stock (MTS) production mode on the basis of the given historical information. IDP also refers to the determination of the designs to be produced in the MTS mode to support the selected customer orders subject to a constraint on the total number of inventory designs that can be chosen based on known potentially large set of inventory. The orders that are not included in the MTS mode are served by Make-To-Order (MTO) production mode by default. The orders having insufficient planned production being planned to be served in the MTS mode are assumed to be satisfied by an alternate longer-cycle-time sourcing method (e.g., MTO production, outsourcing) and incur a shortage penalty. The fluctuations in demand leading to the surplus production volume, which remains unutilized is supposed to incur a penalty cost i.e. the opportunity cost of reserving production capacity. The planning decisions of MTS orders are designed for the optimized production, in order to accomplish the aimed inventory levels on a monthly or weekly basis, if the variety of designs and their respective orders served by them are known. In the MTS mode the operational scheduling period can be convincingly assumed to be independent for the couple of reasons. Primarily, due to extended cycle times for slab production, and high reliability of production efficiency on sequencing and scheduling of the bottleneck resource, deficiency in one period cannot be regained in the subsequent period without considerable cost. Subsequently, rescheduling to back till a missed order, results in domino effect, which may cause numerous delayed subsequent customer orders. Therefore, in order to overcome this, either rescheduling of the order within the MTO mode is carried out or the order is completed from external sources i.e. by purchasing of on-hand slabs/coils at relatively higher cost. In addition, modern industries are equipped with sufficient capacity for finishing operations, leading to the shorter processing time for the customized finishing. In the proposed work we put forward a model that allocates a predetermined time-independent charge for each deficiency, and shortage of MTS items remaining within the same period. As per the proposed stochastic linear programming model, once the inventory-level decisions i.e. selection of the kind of order, design, and production-level designs are made, the supply and demand uncertainty is resolved, and optimal allocation of inventory is made realizing the customer orders in a second-stage linear program.

IDP Problem Formulation

The proposed IDP problem has a set of potential supply nodes, J={1,2,.....,*m*}, representing the set of design choices, and a set of potential demand nodes, K={1,2,.....,*l*}, representing the different order choices. As per the application rules the allowable allocations of supply and demand are represented by the edges between the supply and demand nodes. The notations used are shown below:

C_j^e = per unit cost of having surplus inventory of design j;

C_k^s = per unit cost of scarcity for order-type k;

G_{jk} = supplementary revenue from cycle time reduction if design j is applied to order-type k;

C_j^p = additional per unit cost of producing design j in the Make to Stock mode;

d_j = binary decision variable representing the decision to stock design j;

ν_k = binary decision variable; $\nu_k = 1$ if order-type k is supplied from inventory, and 0 otherwise;

c = maximum number of permitted design choices;

W_j = production /procurement planned for design j;

O_{jk} = quantity of order-type k supplied by design j;

u_{jk} = incidence parameter; $u_{jk} = 1$ if design j can be applied to order k and 0 otherwise;

s_k = shortage for order-type k;

e_j = surplus production of design j;

Y_j = random yield rate for design j;

X_k = random demand for order-type k;

ε = random vector with yields, Y_j and demands, X_k, as components.

ξ and ψ represents the set of positive integers and positive real numbers respectively. Similarly ψ represents the set of all real numbers, and A = {0, 1} is the binary set of variables. The domains of the problem parameter are:

$$c \in \zeta\, C_j^{\beta} \in \psi, (C_k^{\beta}, G_{jk}, C_k^{p}, W, u_{jk}, S_k, e_j) \in \psi, and\ (d_j \nu_k) \in A$$

the random vector ε has support $\Theta \subseteq \psi^{l+m}$, probability distribution P, and finite first moments, Ж.

Since IDP is formulated as two-stage stochastic integer program, the first stage corresponds to design and order choices, $d \in A^m$ and $\nu \in A^l$, and the planned production vector, $W \in \psi^m$. The production cost incurred during the

first period is denoted as, $\sum_{j=1}^{m} C_j^p u_j$. The production cost in the second stage is $\sum_{j=1}^{m} C_j^e e_j$ for excess production and for production scarcities a cost $\sum_{j=1}^{m} C_k^s s_k$ is incurred.

The reduction in the cycle time due to matching designs with demand results in the total additional revenue equal to $\sum_{j=1}^{m} \sum_{k=1}^{l} G_{jk} O_{jk}$. The complete problem, assuming a *risk-neutral* firm can be expressed as:

$$Max\{F = -C^p W + R(d, \nu, W)\} \quad \ldots (1)$$

Subject to

$$\sum_{j=1}^{m} d_j \le c \quad \ldots (2)$$

$$d \in A^m,\ \nu \in A^l,\ W \ge 0 \quad \ldots (3)$$

Where, $R(d, \nu, W, \varepsilon)$ is known as the recourse function. It is the expected additional revenue earned, net of any shortage/overage costs, acquiring from the inventory allocation decisions. In reality $R(d, \nu, W) = E_\varepsilon[R(d, \nu, W, \varepsilon)]$ where $R(d, \nu, W, \varepsilon)$ is defined by:

$$R(d, \nu, W, \varepsilon) = \max \left\{ \sum_{j=1}^{m} \sum_{k=1}^{l} G_{jk} O_{jk} - \sum_{j=1}^{m} C_j E_j - \sum_{k=1}^{l} C_k S_k \right\} \quad \ldots (4)$$

Subject to,

$$\sum_{k=1}^{l} u_{jk} O_{jk} + E_j = Y_j W_j \quad \forall j, \quad \ldots (5)$$

$$\sum_{j=1}^{m} u_{jk} O_{jk} + s_k = X_k \nu_k \quad \forall k, \quad \ldots (6)$$

$$O_{jk} \le X_k d_j,\ \forall (j,k), \quad \ldots (7)$$

$$O_{jk} \ge 0,\ E_j \ge 0,\ \forall (j,k) \quad \ldots (8)$$

Equation (1)-(3) representing the complete problem is feasible for any (d, W), due to positive linear basis provided by (e, s) in constraints (5)-(6), and the fact that $Y_j \ge 0, X_k \ge 0, \forall (j,k)$. In addition, randomness takes place only in the R.H.S. of constraints (5)-(8) and the coefficients of second stage are deterministic. The inventory is measured in tons and $O_{jk} S_k$ is treated as continuous variable. The total production in MTS mode accounts for less than the half of the plants capacity and as there are substantial production efficiencies associated with the high volume MTS mode, due to this reason, there is no upper bound on the variables W_j.

Some assumptions are assumed regarding the objective function coefficients such as non-negativity of the shortage costs. The insignificant revenues are such that if for some (j, k), $u_{jk} = 0$ then $G_{jk} = 0$ as well. In addition, $G_{jk} \ge \max\{-C_k^s - C_j^e, 0\}, \forall (j,k)$, i.e., it is never beneficial to select not to allocate accessible supply of design j to order k if $u_{jk} = 1$ for some j. The first stage procurement cost is assumed as $C_j^p + C_j^e > 0$, since otherwise it is insignificantly optimal to produce an infinite quantity of design j, and that for each design j, there is an order-type k such that $C_k^s + G_{jk} > C_j^p$, since otherwise it is optimal to bring to a halt producing designs j completely.

Equation (2) representing the storage cell constraint plays a role analogous to fixed costs, through the implied opportunity cost, linked with not selecting one of the other potential designs. Equation (5) and (7) implies that optimal production level $W_j^* = 0$, if $d_j = 0$ since otherwise, $E_j > 0$ and needless surplus costs are incurred with no superfluous rewards. Constraints (6) and (7) implies that $\nu_k^* = 0$, if all d_j for which $u_{jk} = 1$ are zero.

4. Overview of ESPC Algorithm

A. Endosymbiotic Algorithm

Endosymbiotic Evolutionary Algorithm (EEA) was proposed by [23]. A symbiotic evolutionary algorithm is inspired by the biological co-evolution that is a series of reciprocal changes in two or more cooperative interacting species. The EEA constructs, and maintains a balancing population (Pop-B), and sequencing population (Pop-S) like the existing symbiotic algorithm. Pop-B and Pop-S consists of symbionts that are the individuals representing work assignment to stations, and model sequences respectively. Each of the individuals becomes a partial solution to the problem being solved. EEA maintains another population Pop-BS that consists of endosymbionts. An endosymbiont is an individual formed by the integration of the two types of symbionts, so that it becomes an entire solution representing a combination of work assignment and model sequence. Indeed, Pop-BS represents the process of forming eukaryotes from prokaryotes.

The Endosymbiotic algorithm intends to replicate the natural process of Endosymbiotic evolution. The theory of Endosymbiotic algorithm was first proposed by [22]. The author provides an explanation for the evolution process of eukaryotes from prokaryotes in which the simple structured prokaryotes enter into a larger host prokaryote, and start living together in symbiosis and evolve to a eukaryote. EEA incorporates an evolutionary strategy replicating the Endosymbiotic process embedded in an existing symbiotic evolutionary algorithm.

B. Psychoclonal Algorithm

The Psychoclonal algorithm enjoys the flavour of Maslow's need hierarchy theory [24] and Theory of clonal selection [25]. Maslow's need hierarchy theory helps in constraints satisfaction by assessing the

antibodies formed at each step. The clonal part helps in the somatic maturation of antibodies. Need Hierarchy Theory [24] hypothesize that all people posses a set of five needs arranged in hierarchy, from most fundamental or basic survival need to the most sophisticated needs of self-actualization. According to this theory, one can move to upper strata of hierarchy if the lower levels of needs are satisfied.

Clone selection explains the response of the immune system, when a non-self antigenic pattern is recognized by a B-cell. Antigen (Ag) stimulates the B-cell to proliferate and mature into terminal Antibody (Ab) (non-dividing) secreting cells, known as plasma cells. The cells divide themselves (no crossover) to generate clones. During reproduction, the B-cells progenies undergo a hypermutation process that together with the strong selective pressure, results in B-cells with an antigenic receptor presenting higher affinities than with the selective antigen. B-cells, in addition differentiate into long-lived B memory cells with a long-life span. These memory cells are pre-eminent in future responses to the same antigenic pattern, or a similar one. The aforementioned, process of clonal, proliferation, and affinity maturation is schematically shown in Figure 1 ([20], and [26]).

Nomenclature

Ab : Set of Antibodies available.
Ag : Set of Antigens available.
Ab_d : Set of the new Ab's that will replace R_c amount of the lower affinity Ab's from Ab.
$Ab_{k,n}$: Ab's from Ab with highest affinities.
Ag_m : Population of m Ag's.
R_k : Population of N_c clones generated from $Ab_{k,n}$.
R_k^* : The population after hypermutation.
BR^* : Best repertoire.
$\hat{A}$: Vector containing values of objective function g (.) as the affinity of all Ab's
$\hat{A}^*$: Vector containing values of antigenic affinity for matured clones. In relation to the antigen, Ag_j
N : The total number of antibodies
N_c : The total number of clones generated for each of the Ag's = $\sum_{i=1}^{n} R\ (\beta . N)$, i=1,2,..n.
R (.) : Operator that rounds its argument toward the closest integer.
β : Multiplying factor
POP_{ij} : Population set of constrained satisfied Ab's.
PAB_{ij} : Population of randomly generated *Ab's*.
PAG_{ij} : Population of randomly generated Ag's.
S : Number of bits in eukaryote.

The flow of the ESPC algorithm has been shown in the Figure 2. The detailed steps of the algorithm have been discussed in the next sub-section.

Steps of the ESPC Algorithm

Need Level I:
Physiological needs: In optimization, this corresponds to the generation of possible sequences based upon the problem environment.
For each cell; PAB_{ij} and PAG_{ij} has been generated randomly, PAB_{ij} is a 2D structure of toroid grid containing the generated set of Ab's. PAG_{ij} is also a 3x3 matrix of randomly generated constraints is Ag's. A set of eukaryotes with satisfied constraints are generated randomly or based on certain rules are stored in POP_{ij} matrix.

Need Level II:
Safety needs: The safety needs has to do with physical and physiological safety from external threats to our well-beings. An external threat in the engineering perspective corresponds to constraints imposed on the problem. This is where evolution of a particular entity or candidate solution is carried out.

Here, new Ab is produced by cooperation between PAB_{ij} and PAG_{ij}. Calculate the affinity vector ($\hat{A}$) of the generated Ab. Randomly select a population from POPij and compare it with newly generated Ab. If the Selected eukaryote *from* POP_{ij} *has* $\hat{A}$ greater than that of generated *Ab* then it will update POP_{ij} toroidal matrix else algorithm will move to improve the quality of *Ab* by cloning and that of eukaryote by carrying out reciprocal changes in it.

Need level III:
Social needs: In engineering this refers to the selection of the candidate solution and the term social reflects the interaction between candidate solutions.

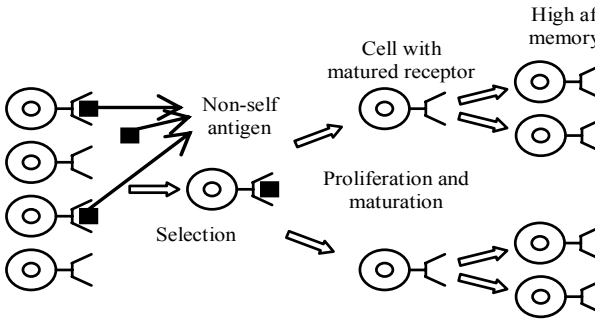

Figure 1: Process of Clonal selection, proliferation and affinity maturation

The selected antibody is cloned again by assigning cooperation between PAB_{ij} and PAG_{ij} and proportionally to the $\hat{A}$, generating a repertoire R_k of clones (higher the antigenic affinity, the higher the number of clones generated for each selected *Ab*).
Need level IV:

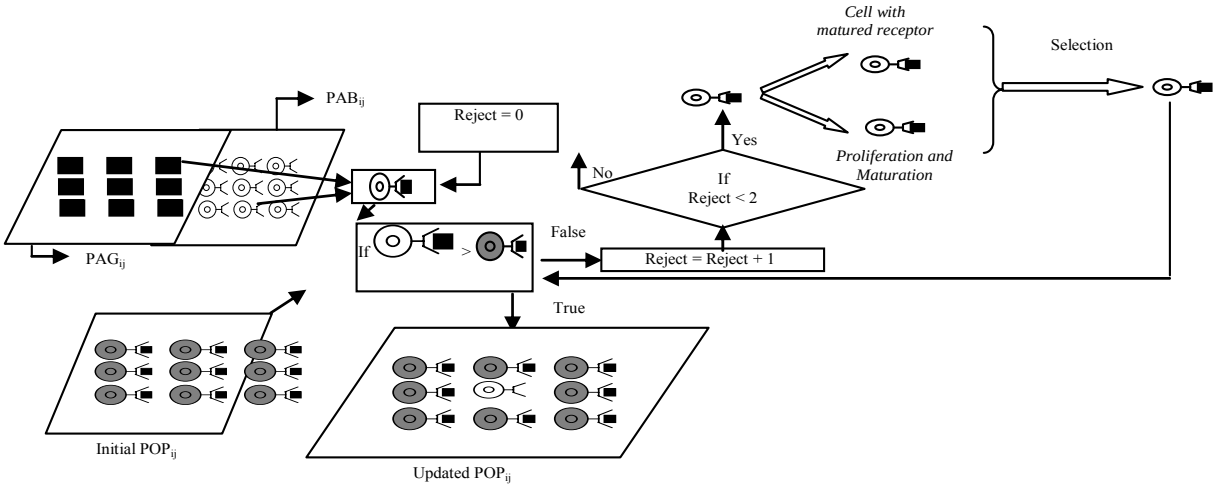

Figure 2: Flow diagram of Endosymbiotic-Psychoclonal Algorithm

Growth needs: Here, candidate solutions diversify to extend the search-space. This movement towards local optima is the basic mechanism of every evolutionary technique e.g. crossover and mutation in GA.

Set R_k is submitted for hypermutation, inversely proportional to the vector affinity (Â), generating a population R_k^* of matured clones (the higher the affinity, the smaller the mutation rate). If the solution of R_k^* don't improve after fixed number of iteration (Reject is taken 2 in this paper) then selected Ab is edited using receptor editing.

After satisfaction of need level IV, vector affinity (Â) of the matured clones R_k^* is evaluated and the best repertoire (BR^*) is passed through Need level III.
Need level V:

Self-actualisation needs: Self-actualization needs are unique and they can never be fully satisfied or fulfilled. This is very true for any optimization problem as we always concentrate on finding near optimal solution rather than the global-optima. According to theory, the more self-actualization needs are fulfilled, the stronger they become.

With the number of generation the solution quality of POP_{ij} goes on improving, when the solution quality stops improving, the algorithm is supposed to achieve self-actualisation needs. In ideal condition self-actualization is achieved at optimal solution. As mentioned above, this level becomes stronger and stronger after a number of generations. Thus, the process repeats till N=Ngen (maximum number of generation).

5. Numerical Experiment

In this study, the authors have proposed an ESPC algorithm to solve an inventory deployment problem. The dataset for different type of scenarios have been randomly generated using the information provided in Denton and Gupta (2004). The coefficients for additional revenues, shortage costs, and excess costs are all uniformly distributed as (1, 4). Additional procurement costs C^P are assumed to be the same for all supply nodes and are fixed at one. The detail information regarding, demand for different order-type, yield rate for different design, Normal probability plot, and Histogram showing the deviation from the normality of the dataset can be obtain from www.geocities.com/gurukul007/inventorydata.pdf.
The ESPC algorithm has been applied on the generated

1	0	1	0	0	0	1	1	0	1

Figure 3: An example of Antibody

data. To initiate the working of the proposed algorithm, initial toroid matrix i.e. POP_{ij} matrix consisting of eukerates have been generated. The eukerates of toroid matrix can be generated randomly or based on some rules. The toroid matrix is generated *i.e.* POP_{ij} matrix containing feasible solution with all constraints satisfied. PAG_{ij} matrix consists *Ag's*, in our case these are the constraints represented by Equation (2)-(8) in section 3. Then PAB_{ij} matrix is generated consisting of *Ab's* viz. candidate solution. The *Ag's* are attacked on *Ab's* randomly i.e. constraints are selected randomly and infeasible solutions are traced back into the feasible solution space based on the constraint represented by the *A*. An example of an antibody has been shown in Figure 3 formed after the attack of Ag's. On the basis of Ab generation, the order-type has been secreted as shown in Figure 4. The Vector affinities of the generated Ab's have been calculated using equation (1). The randomly selected Ab from PAB matrix has been

Table 1: Numerical Results for Randomly Generated Dataset

K=25, U_j ~(0.8,1), $\forall$ *j* and D_k ~N(10,2)

		ESPC		GA		SA	
F(P^a)	(c, n, m)	AV	σ	AV	σ	AV	σ
U (0.1,0.3)	(5, 10, 20)	14.676	0.41	12.256	0.81	9.125	0.31
	(5, 10, 30)	0.929	1.88	0.9156	0.95	0.756	1.63
	(5, 20, 30)	7.234	0.59	6.584	0.48	6.452	0.62
	(5, 20, 40)	3.068	0.24	3.124	0.75	1.251	0.34
	(10, 20, 30)	0.864	0.35	0.758	0.25	0.565	0.76
	(10, 20, 40)	3.514	0.57	2.947	0.20	3.125	0.45
	(10, 30, 50)	4.505	2.14	3.210	1.23	2.265	0.65
U (0, 0.4)	(5, 10, 20)	10.266	0.35	10.256	0.23	9.303	0.53
	(5, 10, 30)	2.051	0.023	2.131	0.35	2.015	0.32
	(5, 20, 30)	10.567	2.54	9.154	3.15	9.532	0.14
	(5, 20, 40)	1.080	0.025	0.926	0.023	0.712	0.21
	(10, 20, 30)	4.135	0.11	3.589	0.15	4.021	1.05
	(10, 20, 40)	6.034	0.54	5.121	0.63	0.593	0.92
	(10, 30, 50)	4.838	0.47	4.568	0.83	3.978	0.35
U (10, .25)	(5, 10, 20)	9.035	0.515	9.142	0.61	8.691	0.61
	(5, 10, 30)	2.851	0.41	2.816	0.56	2.563	0.26
	(5, 20, 30)	2.237	0.64	1.915	0.34	1.654	0.25
	(5, 20, 40)	1.909	0.15	1.896	0.11	1.726	0.21
	(10, 20, 30)	7.420	0.24	7.670	0.68	6.840	1.27
	(10, 20, 40)	11.10	0.86	10.26	1.67	10.641	1.25
	(10, 30, 50)	0.135	0.002	0.054	0.004	0.03	0.002

AV = Average; K = No. of scenarios; D = demand; c= Max. no. of permitted design; n = No. of design choices; m = No. of order choices

compared with the eukerates selected from the toroid matrix represented by POP_{ij} matrix. If the vector affinity of selected antibody is greater then the solution selected from the toroid matrix then it will replace the solution else selected Ab is send for cloning carryout at need level III. The hypermutation is carried out on cloned Ab.

The proposed algorithm has deterministic procedures for finding the rate of hypermutation, which is given as:

$$\sigma = \exp(-\delta * \hat{A}) \quad \ldots (13)$$

Where,

σ = Rate of hypermutation

δ = Control factor of decay

After hypermutation, maturation is carried out, by attacking the *Ag's* on the reproitires formed from cloning. Best matured reproitire has been compared with the eukerates, if the solution doesn't improve for a fixed number of iteration (In this research, Reject is set to 2) the antibody is edited using receptor editing. The solution quality in the toroid matrix *i.e.* POP_{ij} improves with the number of iterations satisfying the self actualization needs of Ab's and eukerates.

The proposed algorithm has been coded in MATLAB 5.5 and tested on PIV 1.9 GHz processor. In this paper, the authors have incorporated the merits of Endosymbiotic algorithm into the Psychoclonal algorithm and proposed a new Endosymbiotic Psychoclonal algorithm (ESPC). The ESPC algorithm has faster convergence than the GA (Genetic Algorithm) or SA (Simulated Annealing) based approaches. A prominent feature of the proposed algorithm is its ability to explore different areas of the solution space simultaneously, by breaking initial chromosomes into several populations, which enables it to take cut above the traditional GA's, where genes are blindly divided into two chromosomes. The diversity in the proposed algorithm is ensured by incorporating cooperation and co-evolution among the symbionts.

Endosymbiotic evolution, which is an extension of cooperative or symbiotic evolution, is yet another novel genetic approach that emulates the natural evolution of endosymbionts. An assay of the search strategy adopted by symbiotic evolution reveals that even though different populations cooperate, the distributed search over all the populations might hinder the convergence to good solutions. The proliferation of Endosymbiotic evolutionary algorithm has endowed the search strategy with an effective passage to get by the aforementioned situation. The subsistence of endosymbionts facilitates the exploitation along with the embedded parallel search that results in speedy convergence to better quality solutions.

1	3	1	7	10	1	8	7	3	10	7	1	1	8	7	8	10	3	7	8

Figure 4: Represents Order-type Assignment on Design Based on Ab Generation

The result obtained by ESPC algorithm has been compared with Genetic Algorithm (GA) and Simulated Annealing (SA).The detailed result has been given in Table I. While comparing the proposed ESPC algorithm with that of the GA and SA the crossover probability was set to be 0.6 and the mutation probability was set to 0.1. In SA the initial temperature was set to 200 and the final temperature was 7. From the table it can be seen that the tendency of SA to get entrap in local optima is very high and therefore the average solution quality of SA in most of the cases is less in comparison with ESPC and GA.

The GA in most of the cases gives result near to that of ESPC but when the experimentation on convergence rate of both algorithms is done, it is found that GA got very slow convergence rate. The average number of generations required by GA to reach optimal / near optimal solution is 761 with standard deviation of 67 generations whereas, in ESPC the average number of generation required by algorithm to reach the optimal/near optimal solution is 264 with standard deviation of 32 generations. Thus, ESPC algorithm is around 288 times faster in converging toward the optimal / near optimal solution then GA.

6. Conclusion

In this proposed work a problem pertaining to inventory deployment problem has been addressed using a new optimization approach named Endosymbiotic-Psychoclonal Algorithm. The performance of proposed Endosymbiotic-Psychoclonal algorithm has been tested on computer simulated dataset, the results obtained are found exemplary when the same has been compared with Genetic Algorithm (GA) and Simulated Annealing (SA). Tuning of various parameters of Endosymbiotic-Psychoclonal algorithm has been rigorously carried out and appropriate values have been selected after large number of trial runs.

For future work a robust methodology need to be devised to tackle the large experimentation time required to tune the different parameters of Endosymbiotic Psychoclonal Algorithm. The ESPC algorithm can be applied to solve multi-objective real time problems involving number of constraints. The proposed algorithm also requires to be tested for solving problems from diverse field of manufacturing environment.

References

[1]E. Ignall, and A. Veinott, Optimality of myopic inventory policies for several substitute products, *Management Science*, 15, 1969, pp. 284-304.

[2]D. Sparling, J. Miltenburg, The mixed-model U-line balancing problem, *International Journal of Production Research*, 36 (2), 1998, pp. 485-501.

[3]Y. Bassok, R. Anupindi, and R. Akella, Single period multi-product inventory models with substitution. *Operation Research*, Vol. 47, 2000, pp. 632-642.

[4]F. Chen, D. Zvi, J. K. Ryan, and D. S. Levi, Quantifying the Bullwhip Effect in a Simple Supply Chain: The Impact of Forecasting, Lead Times, and Information. *Management Science*, Vol. 46, No. 3, 2000, pp. 436-444.

[5]U. S. Karmarkar, Convex/stochastic programming and multi location inventory problems. *Naval research Logistics Quarterly,* 26, 1979, pp. 1-19.

[6]L.W. Robinson, Optimal approximate policies in multi-period, multi-location inventory models with transhipments. *Operations Research,* Vol. 38, 1990. pp. 278-295.

[7]J. A. Buzacott, and J.G. Shanthikumar, *Stochastic Models of Manufacturing Systems,* Prentice Hall, Englewood Cliffs, 1993, NJ.

[8]S. M. Disney, M. Naim, and D. R. Towill, "Genetic Algorithm Optimization of a class of Inventory Control Systems, *International Journal of Production Economics*, Vol. 68**,** No. 3, 2000, pp. 259-278.

[9]S.W. Wallace, Solving stochastic programs with network recourse. *Networks*, Vol. 16, 1986, pp. 295-317.

[10]A. Garg, and C.S. Tang, On postponement strategies for product families with multiple points of differentiation. *IIE Transactions*, Vol. 29, 1997, pp. 641–650.

[11]H. L. Lee, and C.S. Tang, Modeling the costs and benefits of delayed product differentiation. *Management Science*, Vol. 43, 1997, pp. 40–53.

[12]G.A. Graman, and M. J. Magazine, An analysis of packaging postponement, in Proceedings of the 1998 *MSOM conference*, *University of Washington School of Business, Seattle, WA*, 1998, pp. 67-72.

[13]D. Gupta, and S. Benjaafar, Make-to-order, Make-to-stock, or delay product differentiation? - A common framework for modeling and analysis. *IIE Transactions*, Vol. 36, 2004, pp. 529-546.

[14]J. M. Swaminathan, and S. R. Tayur, Managing design of assembly sequences for product lines that delay product differentiation. *IIE Transactions*, Vol. 33, 1999, pp. 1015-1027.

[15] B. Denton, and D. Gupta, Strategic inventory deployment in the steel industry, *IIE Transactions*, Vol. 36, 2004, pp. 1083-1097.
[16] G.L. Nemhauser, and L. A. Wolsey, Integer and combinatorial Optimization, J Wiley, New York, 1999, NY.
[17] F. V. Louveaux, and D. Peeters, A dual-based procedure for stochastic facility location. *Operations Research*, Vol. 40, 1992, pp. 564-573.
[18] G. Laporte, F.V. Louveaux, and L. Van Hamme, Exact solution of a stochastic location problem by an integer L-shaped algorithm. *Transportation Science,* Vol. 28, 1994, pp. 95-103.
[19] U. S. Rao, M.S. Jayashankar, and J. Zhang, A multi-product inventory problem with setup costs and downward substitution, working paper, *Carnegie mellon University, Pittsburgh,* PA, 2000.
[20] M. K. Tiwari, Prakash, Kumar, and A. R. Mileham, Determination of an optimal sequence using the Psychoclonal algorithm. *IMechE, Part-B: Journal of Engineering Manufacture,* Vol. 219, 2005, pp. 137-149.
[21] R. K. Singh, P. Kumar, and M. K. Tiwari, Psychoclonal based approach to solve TOC product mix decision problem*. International Journal of Advanced Manufacturing Technology*, 2005, (ISSN: 0268-3768 (Paper) 1433-3015 (Online)).
[22] L. Margulis, Symbiosis in Cell Evolution, *WH Freeman*, *San Francisco*, 1980.
[23] Y. K. Kim, J. Y. Kim, and Y. Kim, An Endosymbiotic evolutionary algorithm for the integration of balancing and sequencing in mixed-model U-lines, *European Journal of Operational Research*, Vol. 168, No. 3, 2006, pp. 838-852
[24] N. R. F. Maier, Psychology in industry, Boston: *Houghton-Mifflin, (Third Edition*), 1965, pp. 417-419.
[25] L. N. De Castro, and F. J. Von Zuben, Artificial Immune Systems: Part I – Basic Theory and Applications, (Tech. Rep. – RT DCA 01/99). *Campinas, SP: State University of Campinas*, Brazil. [On-Line], 1999a.
[26] L. N. De Castro, and J. Timmis, "Artificial Immune Systems: A Novel Paradigm to Pattern Recognition", *In Artificial Neural Networks in Pattern Recognition, SOCO-2002*, (University of Paisley, UK), 2002, pp. 67-84.