Partner Selection and Production-distribution Planning for Optimal Supply Chain Formation

K.L. Mak and W. Su

Abstract — **This paper seeks to develop an effective methodology to solve the partner selection, and production-distribution planning problem that arises in the formation of an optimal supply chain. The methodology includes a mathematical model which describes the characteristics of the integrated supply chain and an efficient genetic search algorithm. The objective is to minimize the total operating cost. Unlike canonical genetic algorithms, individuals of the proposed genetic algorithm have the ability to learn from their ancestors in order to enhance the convergence speed and the quality of the final solution. The performance of the proposed algorithm is evaluated by solving a set of randomly generated problems. Comparison of the results obtained with that of a conventional genetic algorithm clearly shows that the proposed algorithm is superior to the conventional genetic algorithm. The proposed methodology is described in some details in the hope of thus stimulating the use of similar methodology to the solution of other important problems in industrial engineering practice. Suggestions for future work are also included.**

Index Terms **— Partner selection, Production-distribution planning, Supply chain formation, Genetic algorithm**

I. Introduction

Modern manufacturing is increasingly governed by such factors as global competitiveness, dynamic changes in demand, short product manufacturing cycle and customer oriented productions. Indeed, the need for flexibility, efficiency, and quality has imposed a major change in manufacturing industries. The concept of optimal supply chain formation has emerged in response to these challenges that companies face in today's competitive environment. Instead of being independent companies, each focusing on its own business objectives, more and more companies start to cooperate and share information on their capacities, schedules and cost structures. Once a market opportunity has been identified, an optimal supply chain is then formed to compete for contracts which none of the companies could win on its own. When a contract is completed, the supply chain that has been formed shrinks and eventually dissolves. The companies released will become available for the next coming market opportunity. The optimal supply chain is a dynamic alliance of member companies, with each member company contributing its core competence, in order to secure a larger market share and to produce the required product at the lowest operating cost. Hence, the objective of this paper is to develop an effective methodology, which includes a mathematical model and an efficient genetic search algorithm, to solve the partner selection, and production-distribution planning problem that arises in the formation of an optimal supply chain.

A supply chain consists of different business partners, such as raw material suppliers, manufacturers, assembly plants and customers. Therefore, partner selection is very important in establishing a supply chain with competitive advantages. Korhoren [1] and Davis [2] stated that the selection of business partners is an important function for the information management systems of extended virtual enterprises. Talluri et al. [3], Papazoglou et al. [4], and Mikhailov [5] pointed out that the key issue in forming a virtual enterprises is to select agile, competent, and compatible partners. The mathematical model proposed in this paper classifies these business partners into groups according to the product structure rather than the partners' business function, and uses cost as the criteria to select appropriate partners in forming the supply chain.

Production and distribution planning is another key issue in the formation of an optimal supply chain. A Bill of Material (BOM) which describes the structure of a product in terms of its assemblies, subassemblies and basic parts and their relationships is the basis of the planning process. The proposed mathematical model assumes that (1) a single product is produced, (2) the customer demand in each time period is known for the entire planning horizon, (3) the customer's demand has to be satisfied at the end of the planning horizon, (4) each type of raw material can be supplied by more than one supplier and a supplier can supply more than one type of raw material, (5) a basic part can be manufactured by more than one manufacturer and an assembly/subassembly can be assembled by more than one assembly plant, and (6) a manufacturer (assembly plant) can manufacture (assemble) more than one type of basic part (assembly/subassembly). Hence, production and distribution schedules for each selected company must be optimally formulated for producing product to meet all customer requirements at the end of the planning horizon. The mathematical model therefore aims at formulating such schedules to minimize the sum of the operating costs related to production of the product, production set-up, inventory holding and backlogging, transportation, and establishment of

Manuscript received February 28, 2007.

K. L. Mak is a Professor at the Dept. of Industrial and Manufacturing Systems Engineering (IMSE), the University of Hong Kong. (phone: 852-28592582; e-mail: makkl@hkucc.hku.hk).

W. Su is a PhD student at the Dept. of IMSE, HKU (e-mail: suwei@hkusua.hku.hk).

partnership between any two companies, and takes into account the various operating constraints of the supply chain.

Genetic algorithms [6] are adaptive search algorithms which can generate global optimal solutions to linear or non-linear problems, by adopting the concept derived from natural genetics and the evolution theory. Unlike most traditional optimization methods, the technique can explore multiple regions of the solution space simultaneously. The algorithms maintain a population of candidate solutions and mimic the evolutionary process according to the Darwinian principle of the survival of the fittest. Candidate solutions with a good cost performance have a greater chance to survive and reproduce offspring in successive generations by using the genetic operators of selection, crossover, and mutation. As a result, the search process can converge effectively to the most promising regions, and identify the global optimal solution in the solution space. Unlike canonical genetic algorithms (CGA) [7], individuals of the proposed genetic algorithm have the ability to learn from their ancestors in order to enhance the convergence speed and the quality of the final solution. The effectiveness of the algorithm is then evaluated by applying it to solve the partner selection, and production-distribution planning problem. The results obtained confirm that the algorithm proposed in this paper outperforms the canonical genetic algorithm, thus providing a simple, effective and efficient method for optimal supply chain formation.

The remainder of the paper is organized as follows. Section 2 presents the details of the mathematical model developed to describe the characteristics of the optimal supply chain formation problem. Section 3 presents the genetic algorithm with individuals having the ability to learn from their ancestors and Section 4 shows the evaluation of the effectiveness of the proposed methodology as a useful means to formulate optimal partner selection policy and production-distribution plans. Finally, some concluding remarks are given in Section 5.

II. THE MATHEMATICAL MODEL

The problem is formulated as a mixed integer linear program, taking into account the various operating constraints of the system and the costs related to production of the product, production set-up, inventory holding and backlogging, transportation, and establishment of partnership between any two companies. The objective is to minimize the total operating cost involved in partner selection, and production-distribution planning of the product for meeting customer's demand. Details of the proposed mathematical model are described below:

Notation:

t: time identifier;
T: number of time periods in the planning horizon;
n: operational stage identifier
n1: odd operational stage identifier, i.e., n1=1, 3, 5,…;
n2: even operational stage identifier, i.e., n2=2, 4, 6,…;
N: number of levels in the product's BOM structure;
J_n: number of companies in odd operational stage n;
j1, j2, j3, j4, j5, j: identifiers of companies in an odd operational stage;
I_n: number of items in even operational stage n;
i2, i3, i4, i5, i: item identifier;
k: customer identifier;
K: number of customers

Parameters given:

$PC_{j3(n1)i4(n1+1)}$ = unit cost of company j3 in stage n1 for producing item i4 in stage n1+1;

$PS_{j3(n1)i4(n1+1)}$ = set-up cost of company j3 in stage n1 for producing item i4 in stage n1+1;

$RR_{j3(n1)i4(n1+1)}$ = production capacity of company j3 in stage n1 for producing item i4 in stage n1+1;

$TC_{j2(n2-1)i3(n2)j4(n2+1)}$ = unit cost of transporting item i3 in stage n2 to company j4 in stage n2+1 produced by company j2 in stage $n2-1$;

$FC_{j2(n2-1)i3(n2)j4(n2+1)}$ = fixed cost of link between company j2 in stage $n2-1$ and company j4 in stage n2+1 for item i3 in stage n2;

$H^f_{j3(n1)i4(n1+1)}$ = Inventory holding cost of item i4 in stage n1+1 by company j3 in stage n1;

$H^b_{i3(n2)}$ = backlogging cost of item i3 in stage n2;

T = number of periods in planning horizon;

$D_{i(2N)t}$ = forecasted demand for item i in stage 2N in period t;

$W_{i3(n2)i5(n2+even)}$ = produce one unit item i5 in stage n2+even needs W units item i3 in stage n2;

M = a very large number;

Variables:

$$y1_{j3(n1)i4(n1+1)} = \begin{cases} 1 & \text{if company j3 in stage n1 is used to} \\ & \text{produce item i4 in stage n1+1;} \\ 0 & \text{otherwise;} \end{cases}$$

$$y3_{j3(n1)i4(n1+1)t} = \begin{cases} 1 & \text{if } pr_{j3(n1)i4(n1+1)t} > 0; \\ 0 & \text{otherwise;} \end{cases}$$

$$y2_{j2(n2-1)i3(n2)j4(n2+1)} = \begin{cases} 1 & \text{if a link is established between} \\ & \text{company j2 in stage } n2-1 \text{ and} \\ & \text{company j4 in stage } n2+1 \\ & \text{for item i3 in stage n2;} \\ 0 & \text{otherwise;} \end{cases}$$
$n2=2,......,2N-2$;

$tr_{j2(n2-1)i3(n2)j4(n2+1)t}$ = amount of item i3 in stage n2 transported to company j4 in stage n2+1 produced by company j2 in stage $n2-1$ in period t;
$n2=2,......,2N-2$;

$pr_{j3(n1)i4(n1+1)t}$ = amount of item i4 in stage n1+1 produced by company j3 in stage n1+1 in period t;

$tr_{j(2N-1)i(2N)k(2N+1)t}$ = amount of product i in stage 2N delivered to customer k in stage 2N+1 produced by company j in stage $2N-1$ in period t;

$i^{f}_{j3(n1)i4(n1+1)t}$ = amount of items i4 in stage n1+1 hold by company j3 in stage n1 in period t;

$i^{b}_{i3(n2)t}$ = amount of backlogged item i3 in stage n2 in period t;

Hence, the proposed mathematical model has the following form:

Minimize the objective function

$$\sum_{t=1}^{T}\sum_{n1=1}^{2N-1}\sum_{j3=1}^{J_{n1}}\sum_{i4=1}^{J_{n1+1}} pr_{j3(n1)i4(n1+1)t}PC_{j3(n1)i4(n1+1)}$$
$$+\sum_{t=1}^{T}\sum_{n1=1}^{2N-1}\sum_{j3=1}^{J_{n1}}\sum_{i4=1}^{J_{n1+1}} y3_{j3(n1)i4(n1+1)t}PS_{j3(n1)i4(n1+1)}$$
$$+\sum_{t=1}^{T}\sum_{n2=2}^{2N-2}\sum_{j2=1}^{J_{n2-1}}\sum_{i3=1}^{J_{n2}}\sum_{j4=1}^{J_{n2+1}} (tr_{j2(n2-1)i3(n2)j4(n2+1)t}TC_{j2(n2-1)i3(n2)j4(n2+1)})$$
$$+\sum_{t=1}^{T}\sum_{a2=2}^{2N-2}\sum_{j2=1}^{J_{n2-1}}\sum_{i3=1}^{J_{n2}}\sum_{j4=1}^{J_{n2+1}} (y2_{j2(n2-1)i3(n2)j4(n2+1)}FC_{j2(n2-1)i3(n2)j4(n2+1)})$$
$$+\sum_{t=1}^{T}\sum_{n1=1}^{2N-1}\sum_{j3=1}^{J_{n1}}\sum_{i4=1}^{J_{n1+1}} i^{f}_{j3(n1)i4(n1+1)t}H^{f}_{j3(n1)i4(n1+1)}$$
$$+\sum_{t=1}^{T}\sum_{n2=2}^{2N}\sum_{i3=1}^{J_{n2}} i^{b}_{i3(n2)t}H^{b}_{i3(n2)}$$

Subject to the constraints:

$$y1_{j3(n1)i4(n1+1)} \le RR_{j3(n1)i4(n1+1)} \qquad (1)$$

$$\sum_{j3=1}^{J_{n1}} y1_{j3(n1)i4(n1+1)} \ge 1 \qquad (2)$$

if $W_{i3(n2)i5(n2+2)} > 0$

$y2_{j2(n2-1)i3(n2)j4(n2+1)} = y1_{j2(n2-1)i3(n2)} \times y1_{j4(n2+1)i5(n2+2)}$

if $W_{i3(n2)i5(n2+2)} = 0$ and $RR_{j4(n2+1)i5(n2+2)} > 0$ and

$RR_{j4(n2+1)i5'(n2+2)} > 0\ (i5 \ne i5')$ and $W_{i3(n2)i5'(n2+2)} = 0 (i5 \ne i5')$

$$y2_{j2(n2-1)i3(n2)j4(n2+1)} = 0 \qquad n2=2,......,2N-2 \qquad (3)$$

$$y3_{j3(n1)i4(n1+1)t} = \min(pr_{j3(n1)i4(n1+1)t}, y1_{j3(n1)i4(n1+1)}) \qquad (4)$$

$$0 \le pr_{j3(n1)i4(n1+1)t} \le y1_{j3(n1)i4(n1+1)} \times RR_{j3(n1)i4(n1+1)} \qquad (5)$$

if company j3 in stage n1 is not raw material suppliers

$$0 \le pr_{j3(n1)i4(n1+1)t} \le \min(y1_{j3(n1)i4(n1+1)} \times RR_{j3(n1)i4(n1+1)}, M)$$

$$M = \frac{\sum_{j1=1}^{J_{n1-2}} tr_{j1(n1-2)i2(n1-1)j3(n1)t}}{W_{i2(n1-1)i4(n1+1)}} \qquad n1 = 3,......,2N-1 \qquad (6)$$

$$\sum_{t=1}^{T}\sum_{j3=1}^{J_{n1}} pr_{j3(n1)i4(n1+1)t} = \sum_{i=1}^{I_{2N}}((\sum_{t=1}^{T} D_{i(2N)t}) \times W_{i4(n1+1)i(2N)}) \qquad (7)$$

$$\sum_{j4=1}^{J_{n2+1}} tr_{j2(n2-1)i3(n2)j4(n2+1)t} \le pr_{j2(n2-1)i3(n2)t} + i^{f}_{j2(n2-1)i3(n2)(t-1)}$$
$$n2=2,......,2N-2 \qquad (8)$$

$$tr_{j2(n2-1)i3(n2)j4(n2+1)t} \le y2_{j2(n2-1)i3(n2)j4(n2+1)} \times M$$
$$n2=2,......,2N-2 \qquad (9)$$

$$\sum_{k=1}^{K} tr_{j(2N-1)i(2N)k(2N+1)t} \le pr_{j(2N-1)i(2N)t} + i^{f}_{j(2N-1)i(2N)(t-1)} \qquad (10)$$

$$\sum_{t=1}^{T}\sum_{j=1}^{J_{2N-1}}\sum_{k=1}^{K} tr_{j(2N-1)i(2N)k(2N+1)t} = \sum_{t=1}^{T} D_{i(2N)t} \qquad (11)$$

$$\sum_{t=1}^{T_1}\sum_{j=1}^{J_{2N-1}}\sum_{k=1}^{K} tr_{j(2N-1)i(2N)k(2N+1)t} \le \sum_{t=1}^{T_1} D_{i(2N)t}$$
$$T_1 = 1,......,T-1 \text{ and } T \ne 1 \qquad (12)$$

$$i^{f}_{j3(n1)i4(n1+1)t} = i^{f}_{j3(n1)i4(n1+1)(t-1)} + pr_{j3(n1)i4(n1+1)t} - \sum_{j5=1}^{J_{n1+2}} tr_{j3(n1)i4(n1+1)j5(n1+2)t}$$
$$n1 = 1,......,2N-3 \qquad (13)$$

$$i^{f}_{j(2N-1)i(2N)t} = i^{f}_{j(2N-1)i(2N)(t-1)} + pr_{j(2N-1)i(2N)t} - \sum_{k=1}^{K} tr_{j(2N-1)i(2N)k(2N+1)t} \qquad (14)$$

$$i^{f}_{j3(n1)i4(n1+1)(0)} = 0 \qquad (15)$$

$$i^{f}_{j3(n1)i4(n1+1)T} = 0 \qquad (16)$$

$$i^{b}_{i3(n2)t} = \left[\begin{array}{l} \sum_{i=1}^{I_{2N}}((\sum_{t'=1}^{t} D_{i(2N)t'}) \times W_{i3(a2)i(2N)}) \\ -y1_{j2(n2-1)i3(n2)} \times \sum_{t'=1}^{t}\sum_{j2=1}^{J_{n2-1}} pr_{j2(n2-1)i3(n2)t'} \end{array} \right]^{+} \qquad (17)$$

$$i^{b}_{i3(n2)(0)} = 0 \qquad (18)$$

$$i^{b}_{i3(n2)T} = 0 \qquad (19)$$

$$y1_{j3(n1)i4(n1+1)} \in \{0,1\} \qquad (20)$$

$$y3_{j3(n1)i4(n1+1)t} \in \{0,1\} \qquad (21)$$

$$y2_{j2(n2-1)i3(n2)j4(n2+1)} \in \{0,1\}\ n2=2,......,2N-2) \qquad (22)$$

$$i^{f}_{j3(n1)i4(n1+1)t} \ge 0 \qquad (23)$$

$$i^{b}_{i3(n2)t} \ge 0 \qquad (24)$$

$$tr_{j2(n2-1)i3(n2)j4(n2+1)t} \ge 0\ (a2=2,......,2N-2) \qquad (25)$$

$$tr_{j(2N-1)i(2N)k(2N+1)t} \ge 0 \qquad (26)$$

if company j3 in stage n1 is not raw material suppliers

$$\sum_{i4=1}^{I_{n1+1}} y1_{j3(n1)i4(n1+1)} \le 1 \qquad (27)$$

if not specified,

$n1=1,......,2N-1 \quad n2=2,......,2N$

$k=1,......,K \quad i5=1,......,I_{n2+2}$

$i=1,......,I_{2N} \quad i2=1,......,I_{n1-1}$

$i3=1,......,I_{n2} \quad i4=1,......,I_{n1+1}$

$j=1,......,J_{2N-1} \quad j1=1,......,J_{n1-2}$

$j2=1,......,J_{n2-1} \quad j3=1,......,J_{n1}$

$j4=1,......,J_{n2+1} \quad j5=1,......,J_{n1+2}$

The objective function aims at minimizing the total operating cost. Constraint (1) ensures that partners are selected from companies which are capable of producing the item required. Constraint (2) ensures that at least one partner is chosen to produce the item. Constraint (3) shows that the establishment of partnership between two companies. Constraint (4) shows that production set-up cost is incurred only when the production amount is greater than zero. Constraints (5) and (6) are production capacity constraint and BOM constraint. Constraint (7) ensures that the total demand must be met at the end of the planning horizon. Constraints (8)-(12) are transportation constraints. Constraints (13)-(16) are the balance equations and the initial condition of the finished product. Constraints (17)-(19) are the balance equations and the initial condition of backlogged items. Constraints (20)-(22) indicate that y1, y2, and y3 are binary variables. Constraints (23)-(26) indicate that tr and i are nonnegative variables.

III. GENETIC ALGORITHM WITH LEARNING (GAL)

Details of the proposed genetic algorithm are as follows:

Structure of individuals

In this paper, an individual representing a candidate solution for the partner selection, and production-distribution planning problem, has a structure of the form, $\widetilde{X}=(Y,PR,TR)$. The first part, Y, corresponds to a set of variables $y1_{ij}$ (binary numbers) which represent the partner selection decisions. The second and the third parts, PR and TR, correspond to a set of variables pr_{ij} and tr_{ijk} (floating numbers) which represent the production decisions and the distribution planning decisions, respectively.

Step 1: Initialization

The initial population of individuals is generated randomly.

Step 2: Elitist Selection

In any generation, the i^{th} individual, x_i, is selected according to the probability $p=f(x_i)\big/\sum f(x_i)$ where its fitness value $f(x_i)$ = (1/ objective value). The best individual which has been found so far is always passed onto the next generation to ensure convergence of the algorithm.

Step 3: Crossover

A pair of individuals or parents is randomly selected from the population to undergo the uniform crossover operation. A child individual is then produced after the crossover operation. By uniform crossover [8], a crossover mask which has the same length as the individual's structure is created at random and the parity of the bits in the mask indicate which one of the parent individuals will contribute to the formation of the offspring.

Step 4: Mutation

Once the crossover operation has been completed, the mutation operator scans every position of each offspring from left to right, and perturbs its contents randomly according to the specified mutation rate. A random number ranging from 0.00 to 1.00 is generated for each position. A position becomes a mutating position if the random number generated is less than the specified mutation rate. The content of the mutation position is randomly changed.

Step 5: Evaluation and learning operations

The first part of an individual, Y, determines the partner selection relationship. Hence, for each $Y_i \in \Omega_Y$, there exists $PR_j \in \Omega_{PR}$ and $TR_k \in \Omega_{TR}$ such that, among the individuals in the population with Y_i as their first part, the individual $\widetilde{X}(Y_i,PR_j,TR_k)$ leads to the greatest fitness value $f(\widetilde{X}(Y_i,PR_j,TR_k))$. $\widetilde{X}(Y_i,PR_j,TR_k)$ is then kept in a list to provide possible learning opportunity for related individuals in future populations. Unlike the conventional GA which only keeps the best individual that has been found so far, the proposed algorithm also keeps the best individual corresponding to each $Y_i \in \Omega_Y$. The learning operation follows the procedures shown below:

For each individual in the population
if the list is empty
  add $\widetilde{X}(Y_l,PR,TR)$ *to the list*
else
  if does not exist $Y_i = Y_l$ *in the list*
    add $\widetilde{X}(Y_l,PR,TR)$ *to the list*
  else
    if $f(\widetilde{X}(Y_i,PR,TR)) < f(\widetilde{X}(Y_l,PR,TR))$
      replace $\widetilde{X}(Y_i,PR,TR)$ *in the list by* $\widetilde{X}(Y_l,PR,TR)$
    else

$$\widetilde{X}(PR,TR\,|\,Y_l)=\widetilde{X}(PR,TR\,|\,Y_l)+\delta^*(\widetilde{X}(PR,TR\,|\,Y_i)-\widetilde{X}(PR,TR\,|\,Y_l))$$

$$\delta^* = \arg\max_{(\delta\in A)} f(\widetilde{X}(PR,TR\,|\,Y_l)_\delta) \text{ where } A=\{0.1,0.2,0.3,......,0.9\}$$

if $f(\widetilde{X}(PR,TR\,|\,Y_l)_{\delta^*}) \le f(\widetilde{X}(PR,TR\,|\,Y_l))$, set $\delta^*=1$.

Step 6: Termination of the search

Stop when the optimal solution has been found or the maximum number of generations has been reached, else go to step 2.

In steps 3 to 5, a repairing procedure is embedded in each step to ensure that every solution is feasible throughout the search process.

IV. ILLUSTRATIVE EXAMPLE

Figure 1 presents the structure of a product P_1 by using a four-level BOM. C_1, C_2 and C_3 are components, and R_1, R_2 and R_3 are raw materials. The numbers above the squares indicate the amount of raw materials or number of units of the components needed to manufacture one unit of its parent. Figure 2 shows the companies involved in manufacturing the product. Table 1 shows the companies which can supply raw material R_i (i=1,2,3), and manufacture components C_i (i=1,2,3) and the finished product P_1, and summarizes the capacity of each company as well as the costs related to production of the product, production set-up and inventory holding. Table 2 shows the costs related to transportation and establishment of partnership between any two companies. Tables 3 shows inventory backlogging cost. Data in all tables are randomly generated within their respective bounds.

To facilitate a simple illustration of the effectiveness of the proposed methodology, it is assumed that there is only one period in the planning horizon and the demand of product P_1 is 18 units. The proposed genetic search algorithm (GAL) runs 100 iterations with a population size of 100. The crossover rate and the mutation rate are set to be 0.4 and 0.01, respectively. Figure 3 shows the optimal supply chain formed by using the proposed methodology. In figure 3, the number next to each line represents the amount of raw materials supplied, or the number of components (products) manufactured (assembled) by the selected partners. The dash line represents the partnership which has not been established. The total operating cost is equal to $34979. The results are the same as that of GA and ILOG OPL, a commercial software developed to solve mix integer program optimally. All the algorithms are programmed in VC++.net 2003 and run on a Pentium IV 3.2 GHz computer with 512 Ram. The time required to generate the optimal result is 0.2 seconds which is slightly less than those of the other two algorithms.

To evaluate the performance of GAL, 10 test problems are generated on the basis of the problem considered in the illustrative example. These test problems involve 1, 3, 5, 7, and 10 time periods and the corresponding demand patterns are randomly generated as follows:

D_1^1: 18.
D_1^2: 22.
D_3^1: 10,10 and 26.
D_3^2: 21,11 and 25.
D_5^1: 25, 25,6,17 and 4.
D_5^2: 40, 3,13,6 and 24.
D_7^1: 16,2,39,29,22,23 and 5.
D_7^2: 6,17,4,22,33,24 and 20.
D_{10}^1: 40, 3, 13, 6, 24, 11,21,23,30 and 17.
D_{10}^2: 24,21,21,21,18,21,8,10,30 and 7.

CGA and GAL are used to determine the optimal solutions to the test problems. The genetic parameters used in both algorithms are the same as those used in the illustrative example. In each search run, both algorithms run 100 iterations with a population size of 100. ILOG OPL is not used because of the huge amount of computational effort required, e.g., when the number of time periods is 3, the algorithm still cannot converge to the optimal solution after running for 13 hours.

Table 4 summarizes the best and the average of the best solutions obtained by running GAL and CGA 5 times, and the average of the corresponding computation time needed to complete 100 iterations. Figure 4 shows the convergence behaviour of the two algorithms for a typical test case.

The results show that, when the size of the problem is small, e.g., the problem with 1 time period only, both CGA and GAL can quickly generate the “optimal” solution. However, when the number of time period has increased to 3 or above, GAL outperforms CGA in all cases in terms of solution quality although a slightly longer computation time is needed in a few cases. In addition, the convergence speed of GAL is much faster than CGA, especially when the problem size is large.

V. CONCLUSIONS

In this paper, a methodology which consists of a mathematical model and an efficient genetic search algorithm has been proposed to solve the partner selection, and production-distribution planning problem for the formation of optimal supply chain. The mathematical model aims at minimizing the sum of the operating costs related to partner selection, and production-distribution planning of the product and takes into account the various operating constraints of the supply chain. The genetic search algorithm differs from the canonical genetic algorithm in that individuals in a population have the ability to learn from their ancestors. The proposed methodology has been illustrated by using a numerical example. The performance of the proposed genetic search algorithm has also been evaluated by solving a set of randomly generated problems. Results are compared with that of ILOG OPL, a commercial software, and the canonical genetic algorithm. It is clearly shown that, when the size of the problem is small, the proposed genetic search algorithm with learning capability generates the optimal solution. For a large size problem, ILOG OPL cannot be used because of the huge amount of computational effort needed. The proposed algorithm, on the other hand, leads to a lower cost solution than the conventional genetic algorithm. Hence, the proposed methodology is an effective and efficient design tool for optimal supply chain formation, especially when the size of the problem is large.

To improve the computational performance, the possibility of combining the proposed genetic search algorithm with other search method, e.g., Tabu search, to form hybrid algorithms should be investigated. In addition, the convergence behaviour of the proposed algorithm should also be studied.

REFERENCES

[1] Korhoren P., Huttunen K. and Eloranta E. Demand Chain Management in Global Enterprise-information Management View. *Prod Plan Control*. Vol 9.1998.526-531

[2] Davis M. and O’Sullivan D. Systems Design Framework for the Extended Enterprise. *Prod Plan Control*. Vol 10.1999.3-18

[3] Talluri S., Baker R. and Sarkis J. A framework for Designing Efficient Value Chain Networks. *Int J Prod Econ.* Vol 62. 1999. 133-144.

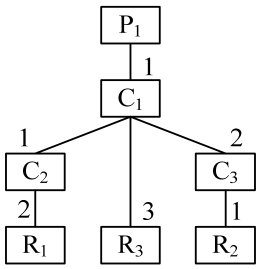
[4] Papazoglou M., Ribbers P. and Tsalgatidou A. Integrated Value Chains and their Applications from a Business and Technology Standpoint. *Decision Support System.* Vol 29. 2000. 323-342

[5] Mikhailov L. Fuzzy Analytical Approach to Partnership Selection in Formation of Virtual Enterprise. *Omega*. Vol 30. 2002. 393-401

[6] Holland J.H. *Adaptation in Natural and Artificial Systems*. Ann Arbor: The University of Michigan.1975

[7] Goldberg D.E. *Genetic Algorithm in Search, Optimization and Machine Learning*. New York: Addison-Wesley. 1989

[8] Syswerda G. Uniform Crossover in Genetic Algorithm. *Proceedings of the Third International Conference on Genetic Algorithm.* 1989. 2-9



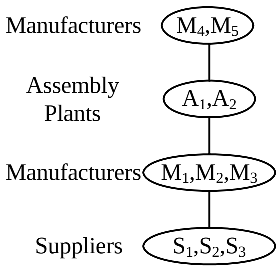


Figure 1 BOM of Product P_1 Figure 2 Groups of Companies

Table1 Capacity, Production Cost, Set-up Cost and Inventory Holding Cost of Each Company

	Capacity	Production Cost	Setup Cost	Holding Cost
$S_1(R_1)$	42	2.3	41.8	1.6
$S_1(R_2)$	38	4.2	75.4	3.1
$S_2(R_1)$	40	3.1	56.2	2.0
$S_2(R_2)$	44	7.5	135.3	4.7
$S_2(R_3)$	72	2.1	37.2	1.4
$S_3(R_3)$	69	3.1	55.5	2.0
$M_1(C_2)$	22	15.3	275.7	10.3
$M_2(C_2)$	17	19.8	357	12.5
$M_2(C_3)$	48	16.2	292.1	10.9
$M_3(C_3)$	44	11.6	209.2	8.6
$M_4(P_1)$	20	332.4	5983.8	246.3
$M_5(P_1)$	24	477.8	8599.6	314.1
$A_1(C_1)$	16	144.1	2593.7	98.2
$A_2(C_1)$	26	162.5	2924.8	106.0

Table2 Transportation Cost and Partnership Establishment Cost Between Companies

	Transport Cost	Partner Cost		Transport Cost	Partner Cost
$S_1(R_1)M_1$	2.5	179.5	$M_1(C_2)A_1$	18.1	770.0
$S_2(R_1)M_1$	4.4	221.9	$M_2(C_2)A_1$	29.5	930.3
$S_1(R_1)M_2$	2.8	224.8	$M_1(C_2)A_2$	21.2	916.9
$S_2(R_1)M_2$	3.3	278.0	$M_2(C_2)A_2$	21.9	1107.8
$S_1(R_2)M_2$	5.4	404.9	$M_2(C_3)A_1$	12.9	1788.5
$S_2(R_2)M_2$	8.5	620.1	$M_3(C_3)A_1$	16.7	1406.8
$S_1(R_2)M_3$	6.0	433.5	$M_2(C_3)A_2$	13.3	1551.7
$S_2(R_2)M_3$	7.4	664.0	$M_3(C_3)A_2$	12.6	1220.5
$S_2(R_3)A_1$	2.9	330.2	$A_1(C_1)M_4$	216.4	6235.1
$S_3(R_3)A_1$	4.8	472.8	$A_2(C_1)M_4$	247.7	6728.3
$S_2(R_3)A_2$	1.6	317.6	$A_1(C_1)M_5$	201.6	6187.4
$S_3(R_3)A_2$	2.5	454.9	$A_2(C_1)M_5$	184.5	6676.8

Table 3 Backlogging Cost of Each Item

	R_1	R_2	R_3	C_1	C_2	C_3	P_1
Cost	3.6	11.7	3.9	305.0	28.7	21.1	623.5

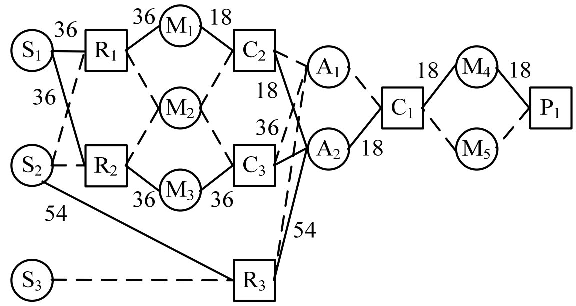


Figure 3 Optimal Supply Chain

Table 4 Results of Test Cases Obtained by CGA and GAL

No.of Periods	Demand Pattern	CGA			GAL		
		Best	Avg.	Time	Best	Avg.	Time
1	D_1^1	34979	34979	9.5	34979	34979	18.6
	D_1^2	43606.8	43609	21.7	43606.8	43609	9.6
3	D_3^1	78278.7	78468.3	20.1	77992.2	78334.7	20.8
	D_3^2	91044.5	91094.3	24.7	90896.6	91049.3	24.6
5	D_5^1	142618.3	143971	36.2	133167.3	133522.6	38.7
	D_5^2	159062.3	159466.2	53.4	158506.6	158712.1	47.3
7	D_7^1	230152.7	230545.3	327	227798.7	228330.3	291.4
	D_7^2	200790	201712.4	95.3	199500.0	200703.0	83.7
10	D_{10}^1	310024.4	310998.9	152.9	307322.6	309914.9	145
	D_{10}^2	357149.8	360624.1	162.1	346696.5	350277.2	180.9

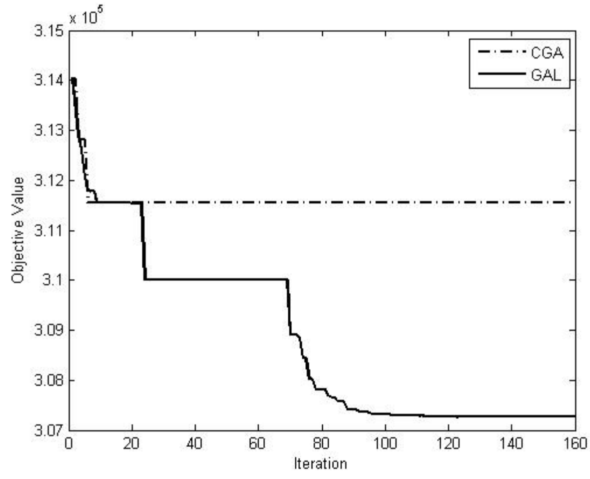


Figure 4 Comparison of Convergence Behaviour of CGA and GAL for Test Case D_{10}^1