INTEGRATION OF GENETIC ALGORITHMS AND SIMULATION FOR STOCKYARD LAYOUT PLANNING

Ramesh Marasini[1] and Nashwan Dawood[2]

Centre for Construction Innovation Research, School of Science and Technology, University of Teesside, Middlesbrough TS1 3BA, UK

This paper describes the development of a process simulation model and integration of the Genetic Algorithms (GA) with the model as optimisation techniques using a case study of stockyard layout planning in precast concrete products industry. Genetic algorithms have been used for two purposes: firstly, to identify clusters of concrete products by analysing sales historical data such that frequently ordered products are grouped together; and secondly, to identify the allocation of the identified clusters to the storage locations. The simulation model was developed in ARENA (simulation software) which evaluates stockyard layout scenarios with different spatial layouts and different allocation of products to storage locations. The evaluation parameters include throughput time for loading and dispatch of the products to service customer orders, queuing and waiting times of lorries and cost of loading and dispatch of the products in the stockyard. The genetic algorithms are used to identify the best allocation of products to storage locations by evaluating the objective function values such as throughput time. The finding from the experiments conducted using a case study are presented and discussed.

Keywords: cluster analysis, genetic algorithm, simulation, stockyard layout

INTRODUCTION

Simulation models are commonly used in the design and evaluation of complex ill - structured systems. Simulation models support to measure and analyse process performance and to develop future process design (Aguilar *et al.*, 1999). In many simulation models, the objective is to measure the output parameters for a given set of inputs and model structure. Trial and error method or several "what-if" analyses are used to select best alternatives. It would be more useful if such simulation models can be used for the identification of inputs or set of inputs to get the desired output. Some of the researchers have highlighted the importance of simulation model input optimisation to get desired outputs. Some examples are: Paul and Chanev, 1998, utilise real coded GA in optimisation of simulation inputs in steelworks simulation model and emphasise a need for optimisation of simulation models; Medaglia *et al.*, 2002 use a fuzzy controller in the design of flow line manufacturing system. In this paper, we will describe the integration of simple genetic algorithms with a process simulation for the optimisation of stockyard layouts.

Genetic algorithms (GA) are search algorithms first developed by John Holland (Holland, 1975) based on natural selection and natural genetics and they are considered robust enough to handle complex problems and have found wide application in diverse areas (Goldberg, 1989). There have been many applications of

[1] R.Marasini@tees.ac.uk
[2] N.N.Dawood@tees.ac.uk

Marasini, R and Dawood, N (2002) Integration of genetic algorithms and simulation for stockyard layout planning. *In:* Greenwood, D (Ed.), *18th Annual ARCOM Conference*, 2-4 September 2002, University of Northumbria. Association of Researchers in Construction Management, Vol. 2, 821-30.

genetic algorithms in civil engineering and construction management, some of them are listed in Li and Love, 2000. This paper describes the development of simulation model to evaluate stockyard layouts for precast concrete industry; discusses the development and integration of simple genetic algorithms and the simulation model. The model was tested and validated using a case study and the results obtained from the study are analysed, discussed and the suitability of the developed model is also presented.

STOCKYARD LAYOUT SIMULATION MODEL

The necessity of stockyard layout planning model and the development of simulation model with an application to precast concrete products industry has been described in Dawood and Marasini, 1999; Marasini and Dawood, 2000 and Marasini *et al.*, 2001. To evaluate and design stockyard layouts, loading and dispatch operations process simulation in Arena 4.0 (Systems Modelling Corporation) was developed. The inputs to the model are production schedules, forecast of sales, vehicle arrival and service distributions, spatial attributes of storage spaces, roads and aisles, order picking policies and allocation of products to storage locations. The model evaluates throughput time to service an order, vehicle-waiting time in the stockyard, space utilisation and cost of loading and dispatch. Using the simulation model several "what-if" scenarios can be analysed.

SIMULATION OPTIMISATION USING GENETIC ALGORITHMS

Simulation optimisation is finding values for the input parameters such that an expected system performance is optimised (Medaglia *et al.*, 2002) and in this study, we have studied the optimisation of allocation of products to storage locations in the stockyard for precast concrete products. Figure 1 shows the optimisation structure of simulation model.

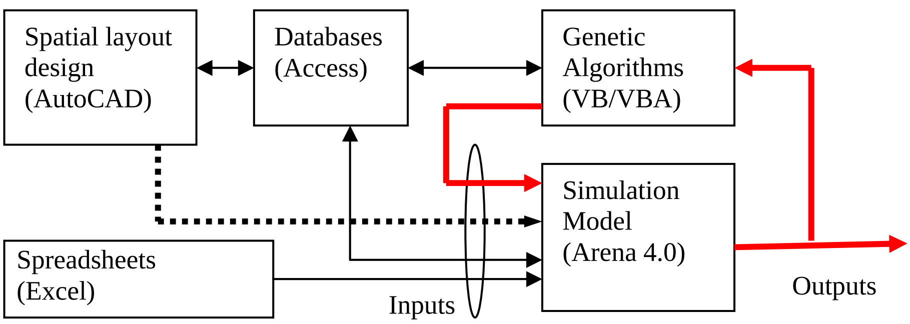


Figure 1: Stockyard layout simulation optimisation framework

In the process of optimisation of the allocation of products to storage locations in the stockyard, genetic algorithms were used for two purposes: clustering i.e. to identify clusters of products that are ordered frequently together and assignment i.e. assignment of developed clusters to storage locations in the stockyard. These are described in the following sections.

Step 1: clustering with genetic algorithms

Cluster analysis deals with the problem of finding subsets of interest called “clusters”, within a set of objects (Hansen and Jaumard, 1997) so that the members of any one subset differ from one another as little as possible, according to a chosen criterion. Details about the clustering techniques, standard terms and definitions can be found in Anderberg, 1973. Some of the previous examples of such applications include Jiang *et al.*,1997; Cowgill *et al.*, 1999; Maulik *et al.*, 2000. The objective of clustering in this study, is to find groups of products, (specifically, the cluster medians) that are demanded frequently together. Historic sales data are utilised for processing and identify clusters within the set of products included in the order history data. Figure 2 shows genetic algorithm procedure used for clustering of concrete products. The terms used in the figures are standard terms used in genetic algorithms (refer Goldberg, 1989).

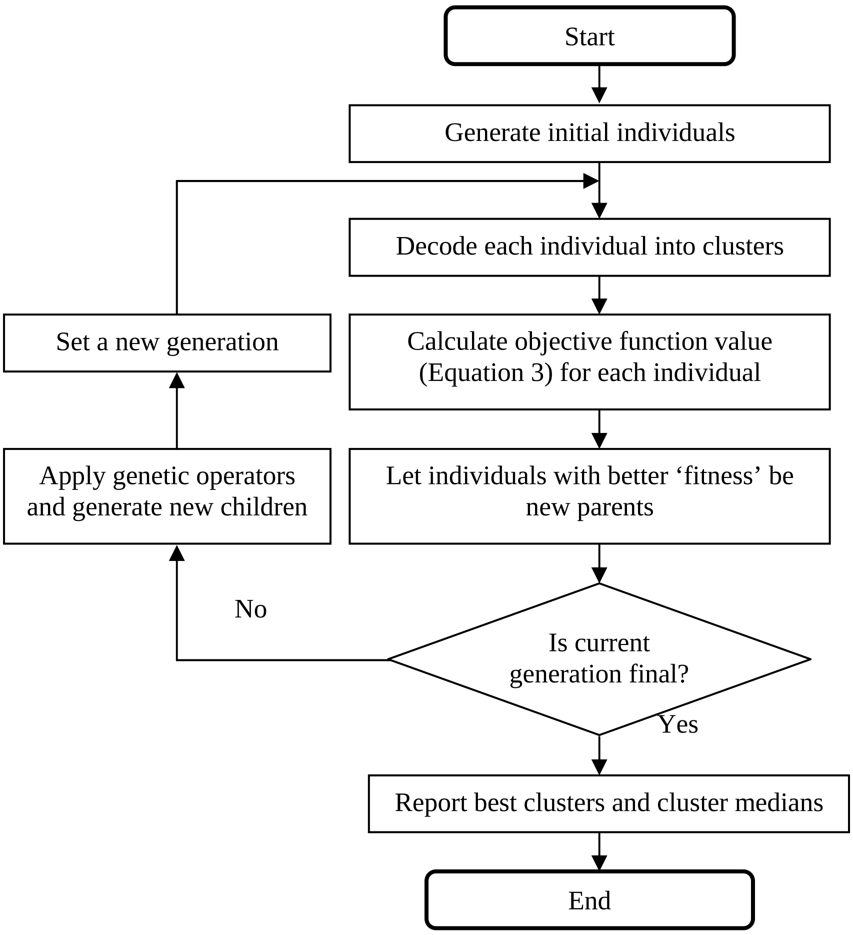


Figure 2: Genetic algorithm (GA) procedure for clustering

Objective Function of Clustering GA
Rosenwein (1994) attempted to group items having similar demand patterns in a warehouse environment and the clustering model utilised by Rosenwein (1994), which was based on Mulvey and Crowder (1979)'s model, is used in this study with different similarity measure. Historic sales data are analysed to identify clusters (a *cluster* is a group of products that are ordered frequently together) and cluster medians (a *cluster median* is an item that is representative of the other items in a cluster). The cluster model is stated as following:

Let us consider a history of $\boldsymbol{Q}$ orders and **I** denote the set of items. Let $\boldsymbol{v_i = (v_1, v_2, \ldots, v_Q)}$ be a vector that summarises the ordering pattern of item $\boldsymbol{i}$ $\boldsymbol{(i \in I)}$. Let $\boldsymbol{q}$-th element of $\boldsymbol{v}_i$ be **1** if item $\boldsymbol{i}$ is required by order $\boldsymbol{q}$, and $\boldsymbol{0}$ otherwise. Thus for any pair of items, the distance $\boldsymbol{w_{ij}}$ between **i** and $\boldsymbol{j}$ is given by:

$$w_{ij} = \sum_{q=1}^{Q} a_{ij} \quad \text{..}\textbf{(1)}$$

where $a_{ij} = 1$, if $v_i^q \,\&\, v_j^q = 1$ ....**(2)**

= 0, otherwise

A high value of $\boldsymbol{w_{ij}}$ indicates that ***i*** and **j** are frequently together.

Again, let $\boldsymbol{x_{ij}}$ be a binary variable such that $\boldsymbol{y_j}$ is ***1*** if item ***j*** is chosen as a cluster median, and ***0*** otherwise. Then cluster analysis model is stated as following:

$$Max \sum_{i \in I} \sum_{j \in I} w_{ij} x_{ij} \quad \text{..}\textbf{(3)}$$

s.t. $\sum_{j \in I} x_{ij}$ = 1 for every $\boldsymbol{i, j \in I}$ ....**(4)**

$$\sum_{j \in I} y_j = p \quad \text{..}\textbf{(5)}$$

$x_{ij} \le y_j$ for every $\boldsymbol{i, j \in I}$ ....**(6)**

Using the cluster model (Equation 3), the cluster medians equal to the number of storage locations were found. The matrices W_{ij} and A_{ij} (matrix that shows how many times any two products were ordered together) are derived by processing sales history. Visual Basics 6.0 and Structured Query Language (SQL) was used to process the data. The matrices are saved in text files and are used while genetic algorithms are run.

Chromosome Representation

In this research, one dimensional array chromosome representing identification code (ID) of concrete products with Integer representation was used to encode the clustering problem. Consider 10 products that are being clustered into 3 clusters. A chromosome is represented as (2 3 1 7 4 5 8 9 6 10). The string is decoded using greedy heuristic so that feasibility conditions are not violated and the objective function value is maximised. The string is decoded as following: First three products (equal to number of clusters required) 2,3, and 1 are initialised as cluster medians. Then unassigned products are assigned one by one to the clusters, which gives maximum objective function value until the size of cluster is reached. The cluster assignment will be:

Cluster 1: 2, 7,4
Cluster 2: 3, 5,9
Cluster 3: 1,8,6,10

As mentioned earlier, the objective is find cluster median that are representative of the group.

Selection Strategy

Reproduction (or selection) is a process in which chromosomes from the mating pool are selected to reproduce as directed by the survival of fittest concept of natural genetic systems. Stochastic remainder sampling without replacement scheme (SRSWR) described in Goldberg, 1989 was used to select the population for reproduction. Elitist strategy was used in developing new generations, which prevents best member of population failing to produce offspring in next generation by copying the best member of each generation into the succeeding generation.

The elitist strategy may increase the speed of domination of population by super individual, but on balance it appears to improve genetic algorithm performance (Davis, 1991).

Crossover
Crossover is a process in which two parent chromosomes are randomly coupled, and each couple of chromosomes partially exchanges information to produce child chromosomes. There are several types of crossover operators (Glodberg, 1989), however, the selection of the operators for a given problem domain depends on the representation schemes.

Partially matched crossover is utilised in this study. In crossover process, the members of newly reproduced chromosomes are mated at random. Two chromosomes are selected at random for reproduction. The selected two chromosomes are aligned and two crossing sites are selected at random. The procedure is represented as following:

P1 = 2 4 5 3 | 1 6 7 | 10 9 8

P2 = 9 8 2 6 | 3 4 5 | 1 7 10

Suppose the random crossover sites are 4, and 7 as shown above with chromosomes P1 and P2. Mapping P1 to P2, the 1 and the 3, the 6 and the 4, the 7 and the 5 are the exchange positions. Mapping P2 to P1, the 3 and the 1, the 4 and the 6, the 5 and the 7 are the exchange positions.

The offspring will be:

O1 = 2 6 7 1 | 3 4 5 | 10 9 8

O2 = 9 8 2 4 | 1 6 7 | 3 5 10

Mutation
Mutation is occasional random alteration of the value of genes in the chromosome with a user-specified probability called mutation rate. This operation is designed either to restore some useful genetic information, which might be lost in the process of crossover operations or to introduce some new information (Mawdesley and Yang, 2000).

If mutation is to occur in a chromosome, then two random numbers between 1 and the string length which give two positions within the chromosome are selected. The genes at these positions are interchanged.

Running of Clustering GA
The crossover and mutation probabilities were used as 0.6 and 0.003 respectively and the clustering GA was run for population of 50 with 2000 generations. The output of clustering process was saved in text files.

STEP 2: ASSIGNMENT OF CLUSTERS TO STORAGE LOCATIONS
i.In the second step, the objective of genetic algorithm is to assign the clusters (developed using the step 1: clustering GA) to the storage locations. The genetic algorithms were integrated with the simulation model using Visual Basics for Applications programming. In this case, the assignment of clusters must be decided such that throughput time for loading and dispatch is minimised. There could be following situations in the assignment of clusters to storage locations.

1. All clusters can be equally assigned to any storage locations i.e. each storage location can accommodate any of the clusters formed.

2. Storage locations have different storage capacities, so that not all clusters can be accommodated in any one of the storage locations.

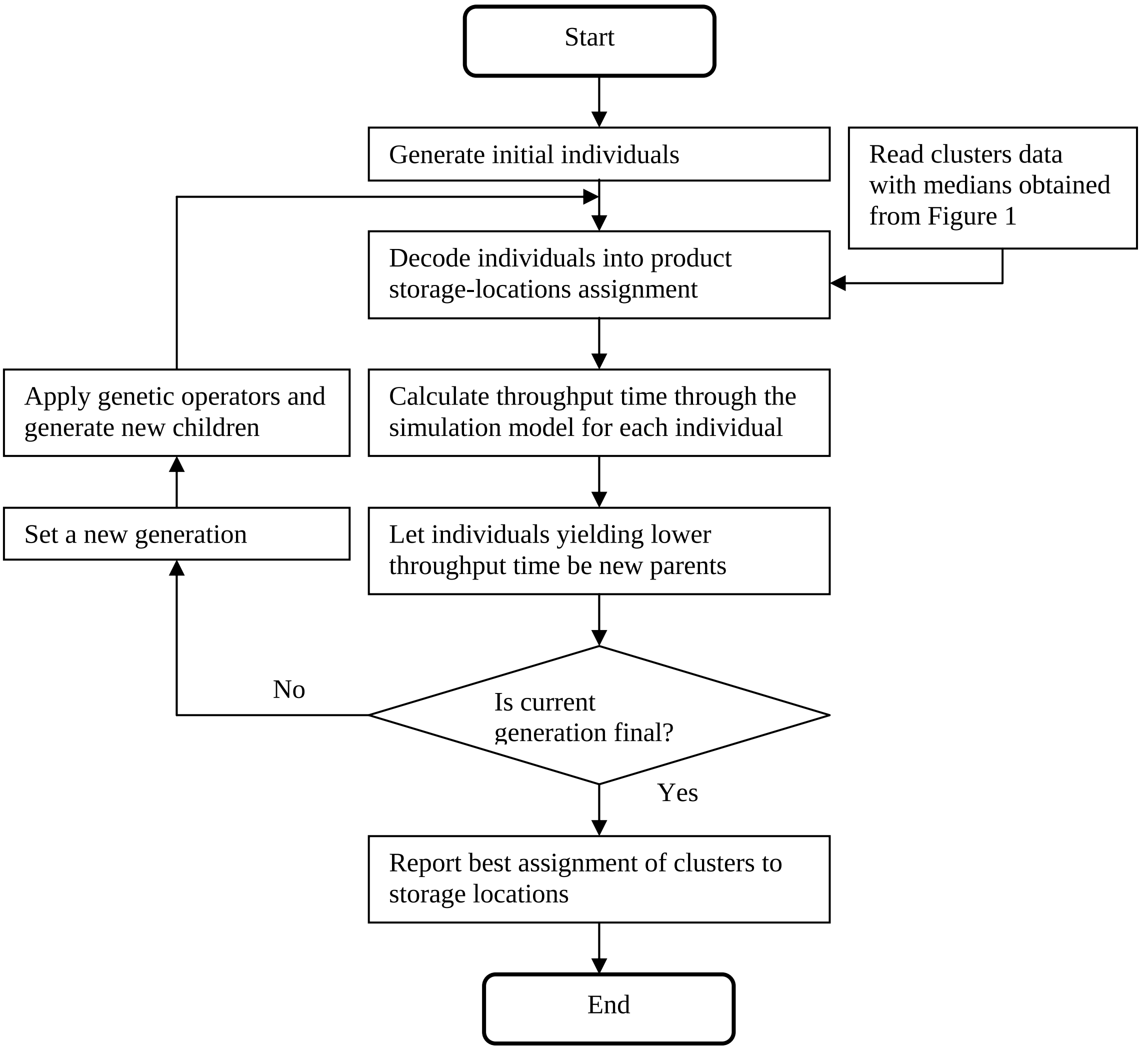


Figure 3: Genetic algorithm for allocation of clusters to storage locations

The first case is simple but deviates from actual scenarios, the reason is that the products have various stock quantities as well as for the similar stock quantity of stock, some products require large storage space and some requiring less. For the given storage space in each storage locations (case ii), it is essential to change the number of products in each cluster such that the space requirements are satisfied. Yet which cluster assignment for a particular location is not known, during clustering process the size cannot be determined. It was, therefore, essential to find best cluster-median product(s) (which is the main essence of cluster model Equation 3) and vary cluster sizes according to products - storage location assignment operation. The clusters are assigned to storage locations using the cluster-medians. Each chromosome representation represents the assignment of cluster medians to the storage locations, and the number of products assigned to the storage locations (and grouped with the median) taking account of the storage space available for the assigned location. Figure 3 depicts the GA procedure to solve the assignment of clusters to storage locations.

Objective (fitness) Function for Assignment GA

Key performance indicators (KPI) for stockyard layout evaluation such as throughput time, cost of loading and dispatch and number of lorries being service in the stockyard can be used for the evaluation of stockyard layouts and to decide with the allocation of products to storage locations. As the values of these variables should be minimised, GA requires the value of objective function be non-negative, the fitness function was modified as follows (equation 7):

$F(x) = C_{max} - Z(x)$ ………………………………………………………. **(7)**

Where, F(x) is the transformed fitness function, C_{max} is the largest value of Z(x) found to this generation and Z(x) represents throughput time or loading and dispatch cost or number of lorries in the stockyard, depending upon the user's choice. In this study throughput time was used as the main objective function.

String Representation and Genetic Operators

The assignment of clusters to storage locations is performed using similar representation and operators used for clustering purpose. But in this case, the string representation itself represents assignment. For example:

Storage Locations: 1 2 3 4 5 6 7 8 9 10

Chromosome: 2 8 5 9 4 3 1 6 10 9

represents that cluster 2 is assigned to location 1, cluster 2 is assigned to location 8 and so on. Individual chromosome is evaluated through a simulation model, which measures throughput time, vehicle queue lengths, space utilisation ratio, and loading and dispatch cost as evaluation parameters. In this study, throughput time has been considered as main evaluation parameter.

Selection strategy and crossover operators used in this case are similar to those were described in earlier section.

Decoding of Chromosome (string)

The chromosome representation is decoded using greedy heuristics (Figure 4). The cluster medians identified by clustering process are assigned to storage locations according to string positions.

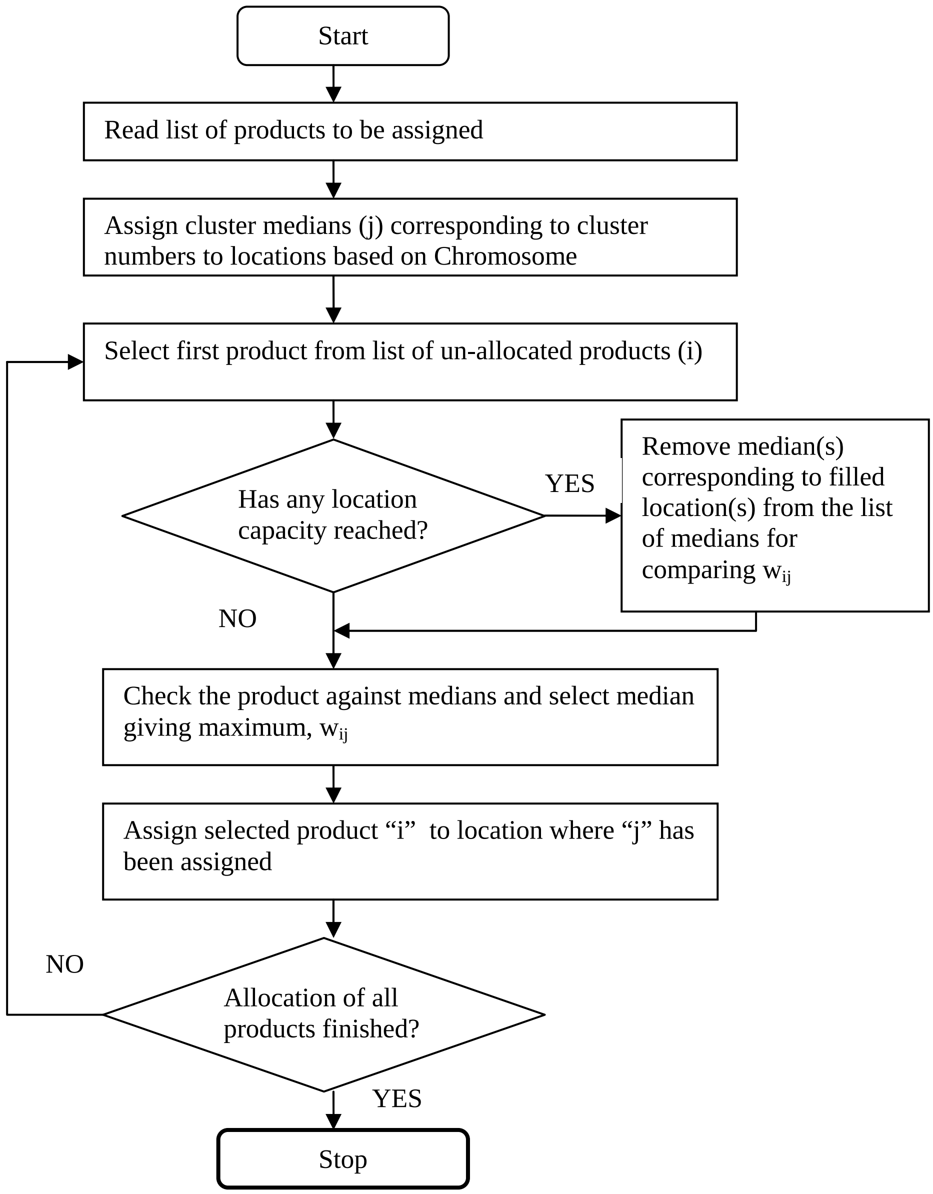
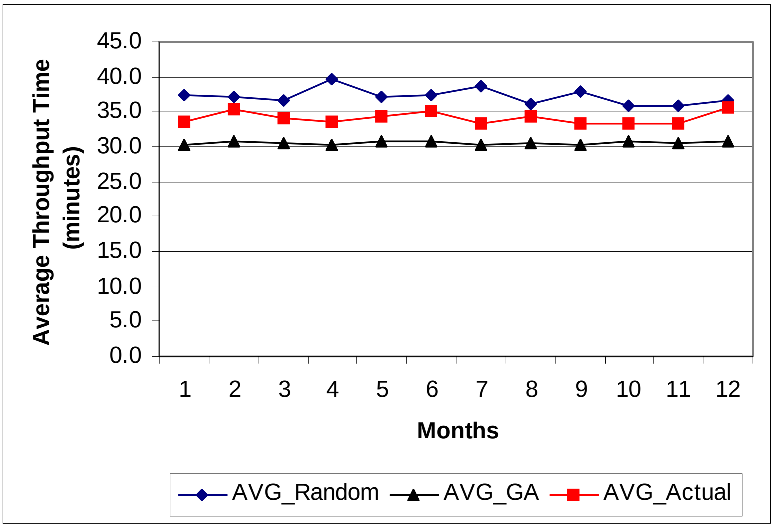


Figure 4: Decoding of cluster assignment to locations

EXPERIMENTATION WITH OPTIMISATION OF ALLOCATION OF CLUSTERS TO STORAGE LOCATIONS

Using the population of 30 and probability of crossover as 0.6 and probability of mutation as 0.0333, genetic algorithms coupled with simulation model were run to identify optimum assignment of clusters to storage locations. The solution converged to a value of throughput time around 24.5~24.75 minutes for the scenarios studied. The model was run with 251 products and area concept of order picking using a case study (details are not included in this paper due to space limitation) and March was used as study month. To compare the allocation of products to storage locations by GA using a case study data the model was run using monthly production and forecast data to evaluate 1-day scenario for each month, for three different scenarios. One scenario was created with random allocation of products, and the second scenario using clustering of products using GA solution, and other with the existing allocation of the products. For each scenarios, 25 replications of the model run were used for each month of the year and average values of throughput time obtained form the analysis have been presented in figure 5.

Figure 5: Variation on average throughput time with different assignment of products to storage locations



It can be seen from the figure 5 that the allocation of products to different locations has considerable effect on throughput time. The throughput time is significantly improved from random allocation to cluster-based allocation identified using genetic algorithms.

SUMMARY AND CONCLUSIONS

A process model for the evaluation of the stockyard layouts for standard precast concrete products was introduced. A framework used to optimise simulation model inputs using genetic algorithms is presented. The genetic algorithms were tested for two purposes: firstly for cluster analysis and secondly, for the assignment of clusters to storage locations. The chromosome representation and genetic operators such as selection, crossover and mutation were described. The assignment type genetic algorithms used for the allocation of clusters to storage locations was optimised by minimising throughput time for loading and dispatch of concrete products. The results have revealed significant improvement in throughput time for loading and dispatch of concrete products using GA-based allocation and this has justified the potential of GA in the optimisation of stochastic simulation models.

REFERENCES

Aguilar, M., Rautert, T. and Pater, A.J.G. 1999. Business process simulation: A fundamental step supporting process centered management, In *Proceedings 1999 Winter Simulation Conference*, eds. P.A. Farrington, *et al.*, pp.1383-1392.

Anderberg, M. R. (1973). *Cluster analysis for applications*. New York: Academic Press.

Cowgill, M.C., Harvey, R.J. and Watson, L.T. (1999). A genetic algorithm approach to cluster analysis. *Computers and Mathematics with Applications*, **37**(1999), pp. 99-106.

Davis, L. (1991). *Handbook of genetic algorithms*. Van Nostrand Reinhold, Newyork.

Dawood, N. and Marasini, R. (1999). Optimisation of stockyard layout for precast building products industry. *Proceedings of Association of Researchers in Construction Management* (ARCOM'99, Sept 15-17), **2**, pp. 445-454.

Deukidis, G.I. and Angelides, M.C. (1994). A framework for integrating artificial intelligence and simulation. *Artificial Intelligence Review*, **8**(1).

Goldberg, D.E. (1989). *Genetic algorithms in search, optimisation, & machine learning*. Addison-wesley: Reading MA.

Hansen, P. and Jaumard, B.(1997). Cluster analysis and mathematical programming. *Mathematical programming*, **79**(1997), pp. 191-215.

Holland, J.H.(1975). *Adaptation in natural and artificial systems: An introductory analysis with applications to Biology, Control and Artificial Intelligence*. MIT Press.

Jiang, J., Wang, J., Chu, X. and Yu, R. (1997). Clustering data using a modified integer genetic algorithm(IGA). *Analytica Chemica Acta*, **354**(1997), pp 263-274.

Li, H. and Love, PED (1998). Site- Level Facilites Layout Using Genetic Algorithms. *Journal of Construction Engineering and Management*, ASCE, **12**(4), pp. 227-231.

Li, H. and Love, PED (2000). Genetic search for solving construction site-level unequal area facility layout problems. *Automation in Construction*, **9**(2000), pp. 217-226.

Marasini, R., Dawood, N. and Hobbs, B.(2001). Stockyard layout planning in precast concrete products industry: A case study and proposed framework. Journal of *Construction Management and Economics*, **19**(4), pp. 365-377.

Marasini, R and Dawood, N. (2000). A simulation approach to optimise stockyard layout: A case study in precast concrete products industry. In Ed Gudnason,G., *Construction Information Technology 2000*,Reykjavik, Iceland, 2, pp. 610-619.

Mawdesley, M.J. and Yang, H. (2000). Genetic algorithms for construction site layout in project planning. *Working paper*, School of Civil Engineering, Univ. of Nottingham.

Maulik, U., and Bandyopadhyay, S. (2000). Genetic algorithm-based clustering technique. *Pattern Recognition*, **33**(2000), pp. 1455-1465.

Medaglia, A.L., Fang, S.C. and Nuttle, HLW (2002). Fuzzy controlled simulation optimisation, *Fuzzy Sets and Systems*, **127**, pp. 65-84.

Michalewicz, Z. (1994). *Genetic Algorithms + Data Structures = Evolution Programs.* Springer-Verlag.

Mulvey, J.M., and Crowder, H. (1979). Cluster analysis: an application of Lagrangean relaxation. *Management Science*, **25**, pp. 329-340.

Paul, R.J. and Chanev, T.S. (1998). Simulation optimisation using a genetic algorithm, *Simulation Practice and Theory,* 6, pp.601-611.

Rosenwein, M.B.(1994). An application of cluster analysis to the problem of locating items within a warehouse. *IIE Transactions*, **26**(1), pp.101-103.