Integrated Intelligent Research (IIR)

International Journal of Computing Algorithm
Volume: 03, Issue: 01 June 2014, Pages: 93-96
ISSN: 2278-2397

Heuristic Algorithm for Constrained 3D Container Loading Problem: A Genetic Approach

S. Jegadeshwari, D. Jaisree
Faculty, MGR University, Chennai

Abstract - This paper presents an heuristic Genetic Algorithm for solving 3-Dimensional Single container packing optimization problem. The 3D container loading problem consists of 'n' number of boxes being to be packed in to a container of standard dimension in such a way to maximize the volume utilization and inturn profit. Furthermore, various practical constraints like box orientation, stack priority, container stability, etc also applied. Boxes to be packed are of various sizes and of heterogeneous shapes. In this research work, several heuristic improvements were proposed over Genetic Algorithm (GA) to solve the container loading problem that significantly improves the search efficiency and to load most of heterogeneous boxes into a container along with the optimal position of loaded boxes, box orientation and boxes to be loaded by satisfying practical constraints. In this module, both the guillotine and non-guillotine moves were allowed. In general, these heuristic GA solutions being substantially better and satisfactory than those obtained by applying heuristics to the bin packing directly.

I. INTRODUCTION

Due to industrial revolution, today most of the industries are focusing towards globalization. Globalized market results in rapid development of international trade and in turn corporations have substantially increased the scope and magnitude of their global production and distribution networks. As a result of global distribution, multinational manufacturing companies have their customers all around the world. So a need arises for those firms to deliver and distribute their goods to customers, warehouses, and to distribution centers all around the world by means of cargo transport. For cargo transport, type and number of containers rented from shipping or from air cargo is a major factor which influences cost of that product and profit margin of the firm. In general, freight rate for each container is sum of fixed cost for using that container and variable cost depends on the total weight of items packed in to the container. Thus it is clear that the product cost increases with increase in cargo cost.

Economical research proved that the survival of a company in this competitive globalized environment will be possible only if that company reduces their product cost without compromising quality. So a necessity arises for multinational companies to reduce their product cost by optimizing many factors influencing cost. It is clear that the major factor influencing distribution cost is cargo cost and can be reduced as least as possible by proper compact packing. During packing of bins, some empty space may be wasted inside the container because all the items will not be of same shape and size. This empty space leads to instability, usage of additional container, airbags, etc. In turn there will be increase in freight cost, transport cost, revenues for exporters, product cost, etc. Thus container selection and packing deserves attention for reducing cost, increase in effectiveness, economical movement and ease of transportation/distribution. So in this research work, container loading problem was taken and solved. On the light of the above, it is necessary to pack the boxes tightly inside the container with less wastage of free space. Some companies were working out manually based on experience but this decision making approach lacks systemization and it is not known how effective these decisions are? Basically no fast algorithm allows an exact solution for this perfect packing. Similarly there is no analytical formulation for loading of maximum number of bins in to the container. So this research work is focused towards heuristic evolutionary algorithms. Among these algorithms, Genetic approaches can solve complex problems in optimization with difficult constraints.

II. LITERATURE SURVEY

In the year 1965, Golmore proposed stack building approach for cutting problem optimization. Historically the first container loading approach was proposed by Christofides, Mingozzi, and Toth in 1979 for solving the liquid loading problem [3]. This algorithm used the dominance criterion and assumed that those liquids will not mix with each other. A procedure for packing boxes into a container using layer-by-layer filling concept was first developed by Gorge and Robinson in 1980. Only a limited number of researches have dealt with container loading problems along with practical constraints with some assumptions. Gehring et al in 1990 developed heuristics by considering practical constraints like stack restrictions, box orientations and container stability along with container loading [7]. Recent research focus diverted towards Meta heuristic algorithms like Tabu search, Genetic Algorithm, simulated annealing, etc were applied to the container loading problem [3]. Bortfeldt and Gehring in 2001 proposed a hybrid GA based on the layer concept for solving the container loading problem with several practical constraints [11]. Interestingly majority of these evolutionary approaches are hybrid / heuristic, i.e. combining GA with other algorithm or by modifying GA. Review of the relevant literature confirms that the evolutionary approach works very well when compared to the traditional optimization procedures and computational time also reduced significantly [7]. As the result of the literature survey, in this research work 3D single container with 'n' number heterogeneous box loading problem using Genetic Algorithm implemented along with many practical constraints.

III. BIN PACKING PROBLEM

Bin-packing has been defined in several different forms such as stock cutting, vehicle loading, air cargo/container loading, scheduling, knapsack problem, etc depending on the

International Journal of Computing Algorithm
Volume: 03, Issue: 01 June 2014, Pages: 93-96
ISSN: 2278-2397

application [16]. Container is a large rectangular box of standard dimension used to store and to transport goods from one location to another location safely. Economic analysis on product cost proved that cost of airbags and freight rate for empty space reduces profit margin. But empty spaces inside the container can be reduced identifying the best mix of boxes and positioning those boxes inside the container. Based on number of containers it is classified as single and multiple containers loading problem. Based on homogeneity of boxes to be loaded, it is classified as homogenous and heterogeneous problem [9]. In this research work, problem related to 3-dimensional single container loading with weakly heterogeneous problem was taken and solved. In this research work, bin shapes may be rectangular, cubical, cylindrical, spherical, etc and bins to be packed are of any dimension and of any shapes, which are discussed in the section 7 in detail. The objective of container loading problem is to fit a fixed number of 'n' items, each weight w_i and volume V_i, into the container of the capacity 'C' without exceeding the capacity of the container and without violating the practical constraints like box orientation, weight, stability, etc. Standard dimension of the container taken for this research is given in the table 1.

Table 1. Container Specification

	Dimension		Dimension
Length	5940 mm	Height	2410 mm
Width	2370 mm	Max. pay load (in Kgs)	18300

IV. HEURISTIC APPROACHES

To solve this type of problems, simple heuristic algorithms used by most of the researchers are First Fit Decreasing Algorithm (FFDA), Best-Fit Decreasing Algorithm (BFDA)[12], Improved First Fit Algorithm, Wall Building Algorithm, Layer-by-Layer Algorithm, Matrix Methods, Tuning Algorithm, Greedy Algorithm, Tree Search Algorithm, Traditional Optimization Algorithms, Stack-Building Algorithm, Guillotine Cutting Algorithms, Cuboids Arrangement Algorithms, etc. However most of these algorithms concentrates on volume maximization and lacks in considering technological and practical constraints. So results obtained by these algorithms are best but unfortunately, the result may not be feasible. In recent years researchers focused towards solving practical constraints and found that evolutionary heuristic algorithms will solve those type problems and solution obtained will be best feasible to pack all bins in to the container [6]. Again most of the heuristic algorithms are time consuming. To find optimal solution by incorporating most of the practical constraints with less computational time, researchers are focused towards evolutionary algorithms like Genetic Algorithm, Tabu Search, Simulated Annealing, etc. Among all this approaches, Genetic Algorithm has become a popular choice of algorithms for numerous difficult optimization problems where conventional methods tend to fail [7]. So in this research work, Heuristic Genetic approach is used for finding optimal loading of bins in to the container by satisfying practical constraints.

V. GENETIC ALGORITHM

Recent revolutions in molecular genetics made clear that the modular organization of genes is highly important for evolution of complexity. Evolutionary concept of Genetic Algorithm (GA) was introduced by John Holland in 1975 at the University of Michigan [1]. First stage in using Genetic Algorithm is initial population generation randomly based on probability logics and each individual in the randomly generated population is represented by chromosomes in biology. The method of random generation may vary from researcher to researcher. Chromosomes have many segments and 'n' number of strings depends on the complexity of the given problem [2]. In this research work, each segment in the chromosomes represents a bin and may contain constraints in the form of strings. Generally population size should be fixed based on the complexity of the problem. So there should be a tradeoff between computational time and number of chromosomes in the population [1]. Second stage is reproduction or crossover between parents in the population. To achieve a good loading pattern that maximizes volume utilization of a container it is necessary to select best two parents from the population. So a need arises to sort parents in some fashion. In order to sort chromosomes, fitness function or evaluation function was formulated by considering practical constraints. Fitness function value was calculated for all parents in the population and sorted based on fitness value. Two point Cross over operator was applied between two randomly selected parents and allowed to reproduces two offspring's. One of the generated offspring inherits best properties from both parent's i.e. best position of packing boxes in to the container. Packing and positioning of bins found may not be optimal in first generation itself and optimal fitness may seems to be stagnating around the optimal point so mutation operator helps to attain best optimal point. Finally generated chromosome has optimal and feasible solution for packing of boxes into the container [1].

Table 2 : GA parameters

Parameter	Value	Parameter	Value
Crossover rate	100	Population size	100
Crossover point	2	Max no of generation	500
Mutation rate	1 %	Genome length	135
Mutation point	1	Min. fitness value	10

In this research work several preliminary experiments were conducted to determine the suitable parameter values and these values shown in the table 2.

VI. MATHEMATICAL MODEL

This section will focus on fitness function development. Assumptions made in building the mathematical model are (i) Origin of the container should be lower most left corner (0,0,and 0) and should be the placement coordinate of first box. (ii) Length, width and height of the container should be along x – axis, y-axis and z-axis respectively. (iii) To be able to relate the placement of different boxes to each other in this research work container is placed in Cartesian coordinate system. (iv) layer by layer filling concept used.

Fitness function for basic single container loading problem [6] can be formulated as Whereas F(x) is the fitness function used to calculate the survival value for the chromosomes. Fitness function is the sum of volume function (total volume occupied by boxes inside the container), placement function (to check

International Journal of Computing Algorithm
Volume: 03, Issue: 01 June 2014, Pages: 93-96
ISSN: 2278-2397

the boundary crossing) and constraint function. Fitness function value ranges from 0 to 100.

$$\text{Maximize } \frac{\sum_{i=1}^{n} (l_i\, w_i\, h_i)}{V} \qquad \text{Eq2}$$

Where l_i is the length of the placed box. (i = 1 to n). w_i is width of the placed box. h_i is height of the placed box. Function C(x) is constraint function. Feasibility of all generated candidate solutions can be guaranteed by satisfying equation C(x). It includes overlapping constraint, stability constraint, weight or load bearing constraint, orientation constraint, shipment priority constraint, stack constraint. If all this constraint satisfied then the function C(x) will return positive value else negative value. It is possible to have a solution with a fitness value greater than zero that is still a valid solution is shown in the equation 4.

$$C(x) = O(x) + S(x) + W(x) + R(x) + P(x)$$

Where as O(x) is overlapping constraint to check interference between boxes. S(x) is stability function to checks stability criteria of packed boxes i.e. width–height ratio, width-length ratio and height-length ratio of boxes and with container will be calculated. W(x) is the weight or load bearing function.

Generally lower weight bins and fragile items should be packed in top layers. R(x) is the orientation function. Box can be rotated along x axis, along y axis, along z axis or the combination of above. Orientation of can be given along with the user input. P(x) is the shipment placement constraint. Boxes should be placed based on the unloading location.

VII. EXPERIMENTAL SETUP

Once this model was decided upon, implementation preceded using Visual Basic (VB) software with MS-Access as database was used. Experiments were conducted using Intel® Pentium (IV) with 2.06 GHz CPU and 1 GB RAM. A number of different modules of classes and objects were developed in VB to model genetic algorithm for solving container loading problem. The one such practical problem taken for implementation is shown in the table 3. When GA starts execution, it gets inputs from user and user defined parameters should be stored in the database. Stored data will be combined to makeup one chromosome of length 135 and length depends on the application. Large size of the search space also adds to the computational complexity when solving any given packing problem and decided based on chosen chromosomal representation. Optimal number of chromosomes in the population will assign to 100. GA configures a few variables such as probability of mutation, probability of crossover, generation size, population size, fitness evaluation assumptions, etc. Chromosomes are allowed to operate over the fitness function and calculated fitness values should be again stored in the same database. Boxes are sorted based on the calculated fitness values. A random method used for selecting parents in this work is roulette wheel selection method and those selected parents are allowed for reproduction. In reproduction stage, 2 – point crossover operator is applied as the length of the chromosome is longer and the points in the chromosomes are selected again at random [1]. At crossover point the strings were exchanged between two parents and as the result two offspring's were generated. Generated offspring may stagnate near optimal point so mutation operator is applied over the offspring. In this work mutation operator changes the orientation of the boxes i.e. length to breadth and vise versa.

Box Type	Box No.	No. of Boxes	Dimensions (cm x cm x cm)	Rotations (x–y– z)	Place ment	Weight 100x(Kg)	Description
CUB	1	10	10 x 10 x 10	1 x 0 x 0	1	2.9	Cast materials
CUB	2	50	20 x 20 x 20	1 x 0 x 0	2	1.32	Wood
RP	3	20	10 x 5 x 2	1 x 1 x 0	1	1.8	Compressed wood
RP	4	70	20 x 10 x 5	1 x 1 x 1	0	1.1	Ceramic Blocks
CY	5	50	50 x 5	1 x 0 x 0	2	0.98	Wood
SR	6	60	20	1 x 1 x 1	0	0.2	Glass boards
FP	7	40	20 x 10 x 5 x 3	1 x 1 x 1	2	0.72	Cardboards
FP	8	50	30 x 20 x 10 x 6	1 x 0 x 0	1	2.03	Gypsum Boards
FC	9	40	20 x 10 x 5	1 x 0 x 0	1	0.9	Cardboards
FC	10	40	10 x 5 x 3	1 x 0 x 0	1	1.4	Gravel Blocks

Table 3. Box Dimensions

$$F(x) = V(x) + P(x) + C(x)$$

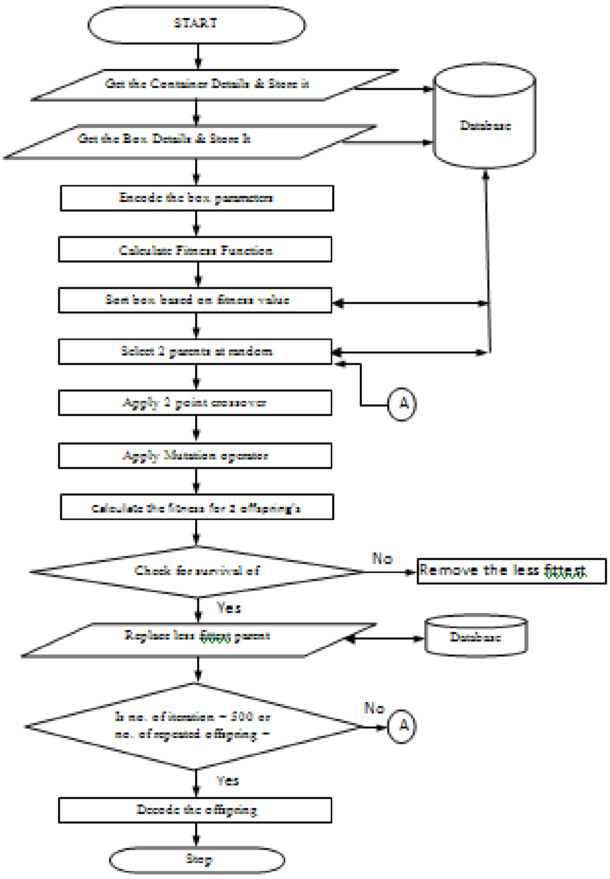

Figure 1. Flow Chart represents the overview of the Research

Fitness values for each and every generated individual offspring (chromosome) in a population were evaluated and offspring with better fitness value is added in to the initial population by replacing the least fitness value parent chromosome. Thus first generation was completed with a new set of best parents for next generation. For the second generation the same procedure was repeated until the number of generation reaches 500 iterations or the continuous 15-generation of offspring with same character. These termination criteria also varied with the application. Final offspring gives the best placement of boxes inside the container and the obtained result may not be optimal result but it will be best feasible result. The overall process of this research work is explained in the figure 1.

VIII. CONCLUSION

The data analyzed so far with this research supports the fact that two point crossover and orientation mutation GA will decrease the execution time of container loading problems with satisfying practical constraints. After iterating many test cases it was found that the factor affecting the computational time are number of bins, type of fitness function, number of constrain violations, size of parent and population. However, a large population needs a huge time for fitness evaluation so the parent size can be reduced as population size increases and vice versa will reduce computational time i.e. population size and parent size should be inversely propositional. So by using GA it will gives best result for any type of complex packing problems.

REFERENCES

[1] D.E. Goldberg, “Genetic Algorithm in Search, Optimization and Machine Learning”, Addison Wesley, 1988.
[2] M.Mitchell, “An Introduction to Genetic Algorithm”, MIT Press, 1996.
[3] E.Hopper and B. Turton, “Application of Genetic Algorithm to Packing Problems – A Review”, Springer Verlag, London, pp 279 -288, 1997.
[4] R. Korf, “A New Algorithm of Optimal Bin Packing”, In proc. AAAI, pages 731-736, 2002.
[5] R. Korf, “A Improved Algorithm for Optimal Bin Packing”, In proc. IJCAI, pages 1252 – 1258, 2003.
[6] D.Pisinger, “Heuristic for Container Loading Problem”, European Journal of Operation Research 141, 292 – 382, 2002.
[7] H.Gehring, A. Bortfeldt, “A Genetic Algorithm for Solving the Container Loading Problem”, International transactions in Operation Research, 44, pages 401 – 418, 1997.
[8] KA Dowsland, EA Herbet, “Using Tree Search Bounds To Enhance A Genetic Algorithm Approach Two Rectangle Packing Problems”, European Journal of Operation Research 168, 390-402, 2004.
[9] S. Martello, D. Pisinger, “The Three Dimensional Bin Packing Problem”, Operation research 48, 256-267, 2000.
[10] EE Bischoff, “Three Dimensional Packing of Items with Limited Load Bearing Strength”, European Journal of Operation Research 168, pages 952-966, 2004.Bortfeldt and H. Gehring, “A Hybrid Genetic Algorithm for Container Loading Problem”, European Journal of Operation Research 131, page 143 -161, 2001.
[11] SG. Christensen and D.M. Rousoe, “Container Loading with Multidrop Constraint”, masters thesis, Informatics and mathematical Modelling, Technical University of Denmark, DTU, Lyngby, 2007. http://www2.imm.dtu.dk/pubdb/p.php?5225.
[12] A.P. Davies and E.E. Bischoff, “Weight Distribution Considerations In Container Loading” European Journal of Operation Research 114, pages 509-527, 1999.
[13] John A George and Jennifer M George, “Packing Different Sized Circles into a Rectangular Container”, European Journal of Operation Research 84, pages 693-712, 1995.
[14] E.K. Burke, M.R. Hyde, “Evolving Bin Packing Heuristic with Genetic Programming”, School of computer science and information technology”, UK. http://cs.nott.ac.uk/~mvh.
[15] Dyckhoff H. A typology of cutting and packing problems. Eur J Oper Res 1990;44:145–59.