Improving Packing Efficiency for Shipping Container

Peeraya Thapatsuwan, Warattapop Chainate and Pupong Pongcharoen

Industrial Engineering Department, Faculty of Engineering, Naresuan University, Pitsanulok, Thailand 65000
E-mail: pupongp@nu.ac.th and pupongp@yahoo.com

Abstract

This present paper focused on the efficiency improvement on the container packing. The modified genetic algorithm (MGA) was developed to find the solution of arranging heterogeneous boxes into containers with a standardised dimension. The proposed MGA included an elitist strategy and flexible arranging sequence (FAS). Five instant datasets were used in the computational experiments aiming to benchmark the proposed GA with the simple GA. The analysis of experimental results demonstrated that the volume utilisations of shipping containers obtained from MGA were improved dramatically especially in the extra large problems.

Keywords: Container packing, Elitist strategy, Genetic algorithm.

1. Introduction

Container packing problem (CPP) is the process of arranging or packing a number of rectangular boxes of products (or items) into a set of containers. Boxes could be in identical size (homogeneous), few different sizes (weakly heterogeneous) or many different types of boxes (strongly heterogeneous) (Bortfeldt, 1994). The efficiency of container packing can be measured in the percent of volume utilisation by minimising the empty space left between boxes in the containers. The container packing problem is usually faced by most global shipping companies using shipping containers of standardised dimensions. It has been reported that the volumes of cargo shipping via Laem Chabang seaport during fiscal year 2006 were 4,123,123 TEUs (Twenty-foot Equivalent Units).

Multiple container packing problem (MCPP) is a combinatorial optimisation problem. Some types of CPP, e.g. heterogeneous boxes, can be classified as NP hard problem, which means that it is often impossible to provide exact solutions within a reasonable time limit. Since the total number of arranging sequences of boxes increases exponentially corresponding to the amount of boxes (*n*) to be arranged and six ways of turning or flipping the box (6^n), the possible solutions for arranging 10 boxes are $(10!) \times 6^{10}$ or 2.19×10^{14} sequences. Metaheuristics (e.g. genetic algorithm) are an alternative approaches particularly for a very large combinatorial optimisation problem since the conventional approaches may have a difficulty on the computational time and resources required (Nagar et al., 1995; Osman and Laporte, 1996).

Many approaches, for instance Wall-building (George and Robinson, 1980), Guillotine Cutting (Morabito and Arenales, 1994), Branch and Bound (Pisinger, 2002), Linear Programming (Beasley, 1985), Tabu

Search (Bortfeldt et al., 2003), Ant Colony Optimisation (Lee et al., 2005) and Genetic Algorithm (Gehring and Bortfeldt, 1997; Pimpawat and Chaiyaratana, 2004), have been used to solve the MCPP. Gehring and Bortfeldt (1997) have applied simple Genetic Algorithm to solve heterogeneous MCPP. Hybrid Genetic Algorithm with greedy heuristic has been proposed by Bortfeldt and Gehring (2001). While Pimpawat and Chaiyaratana (2004) have used a heuristic rule for classifying box sizes into three sub-groups before applying Genetic Algorithm.

The objectives of this present work were to: i) describe the modified Genetic Algorithm (MGA), including elitist strategy, developed for solving the three dimensional container packing problem; ii) propose a flexible arranging sequence embedded within the algorithm; and iii) demonstrate the efficiency of the proposed algorithm.

The remaining sections in this paper are organised as follows. Section 2 reviews the problem statements including assumptions followed by the proposed methodology in section 3. Section 4 then presents the computational experiments based on five instant datasets. Section 5 finally concludes the findings and suggests a further work.

2. Problem Statements and Assumptions

Container packing problem (CPP) may be found in literature under various names, e.g. cutting stock, trim loss, bin packing, container loading, nesting or knapsack problems (Dyckhoff, 1990). CPP can be classified into many categories depending on the criteria such as homo/heterogeneous boxes; packing schemes (Wall-building/ Guillotine cutting) used; *n*-dimensional shapes; rectangular/nonrectangular packing; single/multiple container(s) and so on. Common objective functions used for three dimensional packing are to minimise the length/number of container required for a specified cargo and to maximise the volume of the cargo packed.

Various assumptions have been made in order to simplify, formulate and solve container packing problems. The most common assumptions can be summarised as follows: i) boxes are in rectangular shape with different sizes; ii) boxes must be arranged completely within the container and parallel to its side walls; iii) boxes can be either rotated or non-rotated; iv) priority rules based on packing sequence, e.g. boxes with large size are firstly arranged before the medium and small size boxes respectively (Pimpawat and Chaiyaratana, 2001); v) boxes are stabilised by filling out the empty space left in the container with foam rubber (Pisinger, 2002); vi) boxes on the top are supported by the box beneath; and vii) unlike container loading, weight limitation and distribution may be dismissed in the container packing problem (Davies and Bischoff, 1999).

A general mathematical model for maximising the efficiency of volume usage (V) for multiple container packing problem can be simply formulated as follow:

$$\textit{maximise} \quad V = \sum_{j=1}^{C} \sum_{i=1}^{n} v_i x_{ij} \tag{1}$$

$$\text{subject to} \quad \sum_{j=1}^{C} x_{ij} \le 1 \quad (i = 1, \ldots, n) \tag{2}$$

$$\sum_{i=1}^{n} s_i x_{ij} \le S_{\max} \; (j = 1, \ldots, C) \tag{3}$$

$$s_i,\ v_i,\ S_{max} > 0 \tag{4}$$

where i = box i^{th} from a set of boxes (n)
j = container j^{th} of C containers
s_i = size of box i^{th}
v_i = volume of box i^{th}
S_{max} = container size

The decision variables (x_{ij} 0,1) is used to determine whether to pack box i^{th} into container j^{th}. The n constraints in (2) ensure that each box is packed into a single container only. The C constraints in (3) guarantee that container size (S_{max}) is enough to accommodate all boxes packed into a container. The last constraints in (4) make sure that s_i, v_i and S_{max} are known in advance as positive value.

3. Modified Genetic Algorithm (MGA)

Genetic algorithm (GA) is one of the most famous stochastic search techniques (Goldberg, 1989) that is inspired by the mechanism of natural genetics and natural selection. In this present work, the simple GA (SGA) was modified to include the elitist strategy and a flexible arranging scheme (FAS). The process of the modified GA (MGA) is illustrated in Figure 1. The elitist strategy introduced by Murata et al (1996) is the process of ensuring elite chromosomes surviving into the next generation. A preliminary experimental study aiming to determine the amount (percentage) of elite chromosomes was conducted and it has been found that the amount of elitist strategy embedded into the MGA was recommended in a range between 25%-75% (Samranpun, 2005).

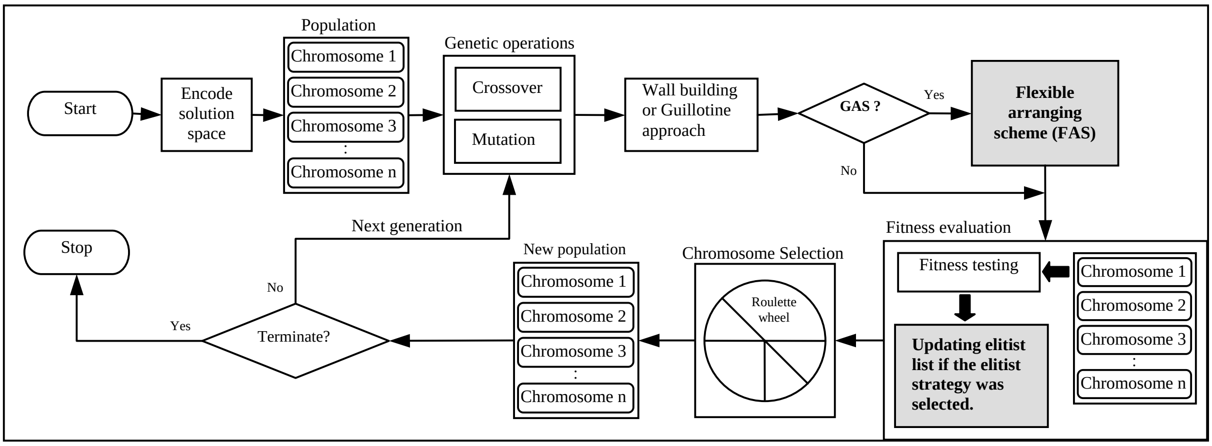

Figure 1 Genetic algorithm for container packing problem.

In this work, the flexible arranging scheme (FAS) is proposed for improving the packing efficiency. Without priority rules, the FAS concept arises when heterogeneous boxes are allowed to be rotated or flipped based on six orientations (see Figure 2). The FAS is the process of considering types of orientation of the successor box based on the height of previous predecessor boxes packed. This means that the orientation of the successor box should be closed to the highest of predecessor boxes in the same row.

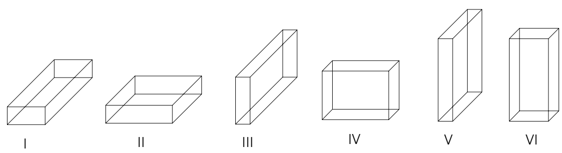

Figure 2 Types of box orientation.

For example, if box (or item) x has been previously packed into a container (see Figure 3), there are six possible ways to pack the next box paralleled to the previous box but there is one orientation that produced smallest margin based on three dimensions. The FAS mechanism was embedded into the

MGA for ensuring that the smallest margins between boxes were left.

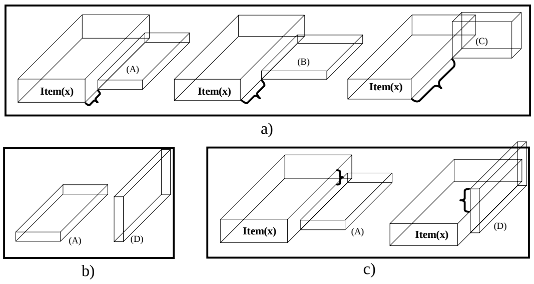

Figure 3 Examples of flexible arranging.

4. Experimental Design and Analysis

An experiment based on a case study was conducted in this present work. Five instant datasets of 100, 500, 1000, 2500 and 5000 three dimensional heterogeneous boxes were randomly created at once and used in the experiment. The length, width and height of each box were randomised in a range of 70-100 cm, 50-80 cm and 30-60 cm, respectively. Boxes to be arranged were not classified into subgroups. A standard size of shipping container with length of 12 metres, width of 2.33 metres and height of 2.35 metres was used in this study. However, the computer simulation program developed allows users to specify these values via a graphic user interface provided.

The objective of this present work was to demonstrate the performance of the proposed modified Genetic Algorithm (MGA) compared with the simple GA (SGA). An experiment was designed using five instant datasets for benchmarking the performance of both algorithms based on the quality of solutions obtained and the computational time spent.

For each dataset, computational experiments were carried with five replications by using different random seed numbers. In each experimental run, the genetic parameters (population size, number of generations and probabilities of crossover and mutation) were specified at 200, 200, 0.5 and 0.15, respectively. The crossover and mutation operators used in this work were cycling crossover and enhanced two box random swap. The setting of these parameters and operators were based on the suggestions resulting from full experimental study and statistical analysis conducted in previous work (Pongcharoen et al., 2001 and 2007). The percentage of elitist strategy used in the MGA was assigned at 75% (Thapatsuwan et al., 2006).

The experimental results obtained from both SGA and MGA were analysed in terms of the volume usage and execution time spent. These experimental results are summarised in Table 1. It can be seen that the experimental results obtained from MGA are always better than those using SGA on various sizes of problem. It was also found that the packing efficiency is increased proportionally when the problem size is enlarged. For example, the best results found from five replications (Best so far) based on extra large problem size (5,000 boxes) obtained from both SGA and MGA were 2,751.6 and 2,105.65 m^3, respectively. This indicated that the proposed MGA can improve the packing efficiency by 23.47%. However, the computational times taken by SGA were 75% quicker than the MGA.

Table 1 Summary of the experimental results obtained from SGA and MGA.

Problem size	Method used	No. of Container	Volume usage (m^3)		% Improve	Time usage (second)
			Average	Best so far		
5000 items	SGA	42	2,783.13	2,751.60	23.47%	4,469
	MGA	33	2,137.81	2,105.65		19,657
2500 items	SGA	21	1,367.63	1,347.14	22.83%	1,852
	MGA	16	1,062.15	1,039.48		27,706
1000 items	SGA	8	529.40	512.56	22.13%	680
	MGA	7	419.64	399.08		3,671
500 items	SGA	4	258.53	248.16	22.98%	337
	MGA	3	199.84	191.12		1,707
100 items	SGA	1	45.97	41.35	14.26%	67
	MGA	1	39.64	35.45		324

5. Conclusions

This paper presents the modified genetic algorithm (MGA) that was developed to solve the container packing problem. The proposed algorithm embedded the elitist strategy and the flexible arranging sequence. Five instant datasets were used in the computational experiments aiming to benchmark the performance of the proposed GA with the simple GA (SGA). The analysis of experimental results demonstrated that the volume utilisations of shipping containers obtained from MGA were improved dramatically especially in the extra large problems compared to those produced by the simple GA. The computational times taken by SGA were however 75% quicker than the MGA. In spite of the growth of interests in using existing methodology, the challenges of industrial packing problems with more practical constraints (e.g. cargo stability and weight distribution) by using new generation of tools or metaheuristics require further research.

References

Beasley, J., (1985). "An exact two-dimensional non-guillotine tree search procedure". *Operations Research*, 33: 49-64.

Bortfeldt, A., (1994). "*A genetic algorithm for the container loading problem"*, Conference on adaptive computing and information processing, London.

Bortfeldt, A. and Gehring, H., (2001). "A hybrid genetic algorithm for the container loading problem". *European Journal of Operational Research*, 131(1): 143-161.

Bortfeldt, A., Gehring, H. and Mack, D., (2003)." A parallel tabu search algorithm for solving the container loading problem". *Parallel Computing*, 29(5): 641-662.

Davies, A.P. and Bischoff, E.E., (1999). "Weight distribution considerations in container loading". *European Journal of Operational Research*, 114(3): 509-527.

Dyckhoff, H., (1990). "A typology of cutting and packing problems". *European Journal of Operational Research*, 44(2): 145-159.

Gehring, H. and Bortfeldt, A., (1997). "A genetic algorithm for solving the container loading problem". *International Transactions in Operational Research,* 4(5-6): 401-418.

George, J.A. and Robinson, D.F., (1980). "A heuristic for packing boxes into a container". *Computers and Operations Research*, 7(3): 147-156.

Goldberg, D.E., (1989). *Genetic Algorithms in Search, Optimisation and Machine*

Learning. Massachusetts, Addison-Wesley.

Lee, Y.H., Kang, J., Ryu, K.R. and Kim, K.H., (2005). "Optimization of container load sequencing by a hybrid of ant colony optimization and tabu search". *Advances in Natural Computation, Pt 2, Proceedings*. 3611**:** 1259-1268.

Morabito, R. and Arenales, M., (1994). "An AND/OR graph approach to the container loading problem". *International Transactions in Operational Research*, 1(1): 59-73.

Murata, T., Ishibuchi, H. and Tanaka, H., (1996). "Genetic algorithms for flowshop scheduling problems". *Computers ind. Engng*, 30(4): 1061-1071.

Nagar, A., Haddock, J. and Heragu, S., (1995). "Multiple and bicriteria scheduling: A literature survey". *European Journal of Operational Research*, 81(1): 88-104.

Osman, I.H. and Laporte, G., (1996). "Metaheuristics: A bibliography". *Annals of Operations Research*, 63(1): 513-623.

Pimpawat, C. and Chaiyaratana, N., (2001). "*Using a co-operative co-evolutionary genetic algorithm to solve a three-dimensional container loading problem"*, The 2001 Congress on Evolutionary Computation (CEC'2001), Seoul, Korea.

Pimpawat, C. and Chaiyaratana, N., (2004). "Three-dimensional container loading using a cooperative co-evolutionary genetic algorithm". *Applied Artificial Intelligence*, 18(7): 581-601.

Pisinger, D., (2002). "Heuristics for the container loading problem". *European Journal of Operational Research*, 141(2): 382-392.

Pongcharoen, P., Chainate, W. and Thapatsuwan, P., (2007). "Exploration of genetic parameters and operators through travelling salesman problem". *ScienceAsia*, 33(2): 215-222.

Pongcharoen, P., Stewardson, D.J., Hicks, C. and Braiden, P.M., (2001). "Applying designed experiments to optimize the performance of genetic algorithms used for scheduling complex products in the capital goods industry". *Journal of Applied Statistic*, 28(3-4): 441-455.

Samranpun, C., (2005). "Applying genetic algorithm for solving the container packing problem", Naresuan University. Master of science**:** 161.

Thapatsuwan, P., Chainate, W. and Pongcharoen, P., (2006). Investigation on the genetic algorithm parameters and heuristic arrangements for container packing problem. *KMITL Science Journal*, 6(2): 274-284.