The HKU Scholars Hub   The University of Hong Kong   香港大學學術庫

Title	**Immunity-based evolutionary algorithm for optimal global container repositioning in liner shipping**
Author(s)	**Wong, EYC; Lau, HYK; Mak, KL**
Citation	**Or Spectrum, 2010, v. 32 n. 3, p. 739-763**
Issued Date	**2010**
URL	**http://hdl.handle.net/10722/145006**
Rights	**Creative Commons: Attribution 3.0 Hong Kong License**

Immunity-based evolutionary algorithm for optimal global container repositioning in liner shipping

Eugene Y. C. Wong · Henry Y. K. Lau · K. L. Mak

Abstract   Global container repositioning in liner shipping has always been a challenging problem in container transportation as the global market in maritime logistics is complex and competitive. Supply and demand are dynamic under the ever changing trade imbalance. A useful computation optimization tool to assist shipping liners on decision making and planning to reposition large quantities of empty containers from surplus countries to deficit regions in a cost effective manner is crucial. A novel immunity-based evolutionary algorithm known as immunity-based evolutionary algorithm (IMEA) is developed to solve the multi-objective container repositioning problems in this research. The algorithm adopts the clonal selection and immune suppression theories to attain the Pareto optimal front. The proposed algorithm was verified with benchmarking functions and compared with four optimization algorithms to assess its diversity and spread. The developed algorithm provides a useful means to solve the problem and assist shipping liners in the global container transportation operations in an optimized and cost effective manner.

Keywords   Liner shipping · Container repositioning · Optimization · Evolutionary algorithms · Artificial immune systems

E. Y. C. Wong (✉) · H. Y. K. Lau · K. L. Mak
The University of Hong Kong, Pokfulam, Hong Kong
e-mail: eugene.yc.wong@hkusua.hku.hk

H. Y. K. Lau
e-mail: hyklau@hku.hk

K. L. Mak
e-mail: makkl@hkucc.hku.hk

1 Introduction

In today competitive environment of global container transportation, an effective logistics flow control system that provides methodologies for equipment planning and optimization is vital to global enterprises such as container liner operators. Under the prevailing atmosphere of trade imbalance, to ensure streamlined operation of container shipment, an effective repositioning for empty containers for timely cargo pickup and delivery through the best route and container stowage will contribute to an optimum and cost efficient supply chain management. Maintaining such a global container flow control system to achieve the above objectives is both complex and resource intensive. An adoption of the traditional Traveling Salesman Problem (TSP) (Finke et al. 1984) and Vehicle Routing Problem (VRP) (Dantzig and Ramser 1959; Golden et al. 2008) for producing empty container repositioning plans is not sufficient to satisfy sophisticated global logistics operations. Often, the development of a repositioning plan requires due consideration of dynamic factors including empty route selections, forecast demand, container stowage, leasing and new built costs, fleet size as well as stock availability. This leads a demand for a multi-objective optimization algorithm for producing such complex container repositioning plans.

To solve the repositioning optimization problem, a novel immunity-based evolutionary algorithm known as immunity-based evolutionary algorithm (IMEA) is developed. In this paper, the IMEA is introduced. It integrates the distinct characteristics of Clonal selection and immune suppression of Artificial Immune Systems (AIS); and genetic theories, with a sorting scheme featuring uniform crossover, multi-point mutation, non-dominance and average harmonic distance sorting to attain the Pareto optimal front in an efficient parallel manner. The global optimal searching performance of the algorithm is evaluated with benchmarking functions and four well-known optimization algorithms. The algorithm is then applied to solve the container repositioning optimization problem. The problem involves the repositioning operations between Europe and Asia Pacific regions with a number of surplus locations, deficit locations, and service routes.

2 Background

2.1 Global container repositioning

The literature on dynamic container allocation have mainly focused on setting up mathematical models for optimizing container repositioning, without any application to real industrial cases. Florez (1986) developed an optimization model on empty container repositioning and leasing. Craninic et al. (1993) developed a mathematical dynamic and stochastic model for empty container allocation. Ting et al. (1996) also developed a mathematical model to minimize empty container allocation costs. More recent literature on containers repositioning include Feng and Chang (2008) and Sun et al. (2009). Relatively few researchers have focused on solving real life container repositioning, one of the multi-objective optimization problems, with the use of the developed repositioning model. Given the characteristics of the developed algorithm

mentioned in the previously, we applied the algorithm to solve the problem of global container repositioning with respect to cost, supply and demand situations, and vessel spaces with a view to assist liner operators to maneuver empty containers in a more efficient and cost effective manner.

2.2 Evolutionary algorithms

Research in evolutionary algorithm lies within the realm of bio-inspired computing, a field with increasing research interest that concerned the use of biological inspired concepts to solve computational problems and real world industrial problems. There have been many representative and novel evolutionary algorithms for solving optimization problems in an efficient manner (Vrugt and Robinson 2007; Chen and Mahfouf 2006; Tan et al. 2001; Zitzler and Thiele 1999; Fonseca and Fleming 1995). One of such evolutionary algorithms—the artificial immune systems-based algorithm—is developed from the inspiration from the human immune system.

AIS is a computational system based on the metaphors of the natural immune system (Timmis and Neal 2000). Recently, an increased interest has been found in formalizing and adapting the theories and underlying mechanisms of the natural immune system to solve complex engineering problems (Luh et al. 2004). Immunological theories have been applied to produce solutions for these problems, including pattern recognition (Garain et al. 2006; White and Garrett 2003; Nicosia et al. 2001), fault detection (Guzella et al. 2007; Taylor and Corne 2003), scheduling (Swiecicka et al. 2006; Coello Coello et al. 2003), and optimization (Omkar et al. 2008; Chen and Mahfouf 2006; Coello Coello and Cortes 2002; Nicosia and Cutello 2002).

In the context of applying AIS to transportation route planning, Keko et al. (2003) adopted the immunizing features of the immune system to a classical genetic algorithm using vaccine inserting method to solve the TSP. Genetic Algorithm (GA) used by Keko was vaccinated to become less susceptible to changing parameters. It further improved the speed of generating the smallest total distance in a TSP problem. A heuristic operator, the Lin–Kerninghan operator, implemented to improve the population with the applied action was considered as vaccination. In his research, the potential in adopting the immunity theory to solve the routing problem had not been fully explored. de Castro and Von Zuben (2000) applied the clonal selection algorithm in immune system to solve a 30-city instance of the TSP. Jozefowiez et al. (2006) applied a Non-dominated Sorting Genetic Algorithm (NSGA II) on VRP with route balancing to optimize the minimization of the difference between the longest route length and the shortest route length. An algorithm, IMEA, is developed in this research adopted the characteristics of the immune system, namely, Clonal selection, somatic mutation, and immune suppression of the immune system.

3 Immunity-based evolutionary algorithm

3.1 Principles and theories

Immunity-based evolutionary algorithm is characterized by the concepts of distributed immune memory, reinforcement learning, self-regulation, and diversity. The

Table 1 Mappings of the entities of the human immune system and those of IMEA

Immune system	IMEA
Antigen	A single objective/multi-objective problem
Antibody	Candidate solution
Immune memory	Archive
Memory cell	Archived solution
Clonal selection	Selection of antibodies contributing with respect to their affinity
Suppression	Eliminating antibodies with close affinity values
Identification of effective antibodies	Affinity calculation
Antibody production and recruitment	Crossover and mutation operator

effectiveness of the immune response to secondary encountering of antigens is enhanced by the presence of memory associated with the first infection, which is capable of producing high affinity antibodies after subsequent encounters. Instead of starting from scratch every time, an intrinsic scheme of reinforcement learning strategy is adopted to ensure a fast and accurate immune response after subsequent infection. Thus antibodies with a high affinity are cloned and stored in the local memory. IMEA also regulates the number of antibodies and increase the diversity of the antibodies in the population through suppression, recruitment, and crossover. The analogies of the IMEA with the human immune system are given in Table 1.

3.1.1 Clonal selection and immune suppression

The concept of domination was adopted for multi-objective optimization search operation where Clonal Selection is used for the exploitation optimal solutions (Burnet 1957). In the algorithm, two solutions are compared on the basis of whether one has dominated the other. The domination is described as: A solution x_1 is said to dominate another solution x_2, if both conditions (1) and (2) are true, where these conditions are:

(1)   Solution x_1 is no worse than x_2 in all objectives, and
(2)   Solution x_1 is strictly better than x_2 in at least one objective (Deb 2001).

The above definition can be applied to any multi-objective algorithm without constrained objectives. A constrained non-dominance scheme is proposed for IMEA. In addition to this definition, Deb proposed that an antibody a_1 dominates an antibody a_2 if any one of the following conditions is true:

1. Antibody a_1 is feasible and antibody a_2 is infeasible;
2. Both solutions are infeasible and a_1 has fewer constraint violations than a_2;
3. Both solutions are infeasible, both solutions have the same number of constraint violations and a_1 dominates a_2 according to the basic definition;
4. Both solutions are feasible and a_1 dominates a_2 according to the basic definition.

The dominance rating of an antibody a_1 is calculated by reckoning the affinity value dominating antibody a_1. Higher dominance rating represents a higher affinity with a better solution. The constrained non-dominance scheme guides the solutions to the global optimal though the above exploitation process.

The search of the optimal solution is further diversified through the exploration stage adopting the immune suppression characteristics. Similar antibodies in the population generated are being suppressed and eliminated. It is performed by estimating the density of the solutions with the use of a crowding degree. The most common ways of calculating the crowding degree are the Crowding Distance and the Harmonic Average Distance. The Crowding Distance is an estimator of density between the solution and its two-nearest neighbor, which is the average distance between the two neighbor solutions (Deb et al. 2000). Harmonic Average Distance estimates the density around a solution. It accounts for all k-nearest neighbors around the solution (Huang et al. 2006). If the distances from the k-nearest neighbors around the solution are d_1, d_2, d_3, ..., d_k, the Harmonic Average Distance is:

$$d = \frac{k}{\frac{1}{d_1} + \frac{1}{d_2} + \frac{1}{d_3} + \cdots + \frac{1}{d_k}} \tag{1}$$

Harmonic average distance is chosen in the algorithm for calculating the diversity affinity instead of the Crowding Distance, as the Crowding Distance does not accurately reflect the density of the solution. The distribution of neighboring solutions may inversely influence the crowding density.

3.2 The algorithm

With the immune characteristics mappings described in Table 1 and the adopted theories, the procedures of IMEA are set up with control parameters including the population size, crossover rate, and mutation rate. The main steps of IMEA are given below.

(1) *Initialization:* Random sets of solutions are generated to form the initial population, known as Population Run Subset (PRS).
(2) *Activation:* After initialization, the solutions in PRS will be calculated for their affinity values. The affinity values refer to the non-dominated affinity (Non-dominated rank) and diversity affinity (Diversity rank).
  • *Non-dominated affinity computation:* It is the calculation of non-dominated rank of each solution. All solutions in PRS will compete with each other to determine its affinity value. First, a non-dominated index which equals zero will be assigned to every solution. If the solution dominates another, the index will remain unchanged. If the solution is dominated by others, its index will be increased by 1, i.e. a solution that dominates 9 solutions and is dominated by 3 solutions has the index of 3. The Non-dominated index is the solution's Non-dominated affinity.

- *Diversity affinity computation:* It refers to the density estimation or crowding degree of the solution: the smaller the affinity value, the more crowded an area the solution is in. Harmonic average distance accounts for all k-nearest neighbor around the solution as shown in Eq. (1).

(3) *Selection:* The whole population is sorted with their respective affinity values. The sorting is based on a primary criterion of non-dominated affinity and a secondary criterion of diversity affinity. Solutions are ranked with the non-dominated affinity. Diversity affinity is further considered if both solutions have the same non-dominated affinity. After sorting the solution set, the PRS are divided into three groups. The best 30% of the PRS is selected to the Optimal Affinity Subset (OAS). The next 40% of solutions are grouped to the Medium Affinity Subset (MAS). The rest of them are arranged to the Worst Affinity Subset (WAS).

(4) *Cloning:* The optimal solution sets in OAS are cloned for crossover and stored for the next iteration. OAS are stored for the next iteration for searching the global optimal. The cloned solutions are called Clone Subset (CLS).

(5) *Mutation:* Solutions in the MAS will undergo mutation with a rate of $1/n$, where n is the number of variables in each solution. The mutation rate $1/n$ means that one variable in the solution will be chosen for mutation. A random variable of the solution is selected for mutation and its mutated variable should be monitored so that the resulting value will remain within the required range. The selected variable is multiplied with a random number ranging from 0.8 to 1.2. The mutated solutions form Mutation Subset (MUS).

(6) *Population suppression:* Solutions in the WAS will undergo population suppression. The 10% of solutions in PRS with the worst affinity value are discarded. The solution with a higher crowding degree has a lower ranking in the diversity affinity. Solutions with the same non-dominated affinity but lower diversity affinity are ranked lower in PRS. The solutions with both worst affinity value and higher crowding degree will have a higher chance of being discarded. The process of Population Suppression also gives space for the other diversification operation—Recruitment. The remaining solutions form the Suppression Subset (SUS).

(7) *Recruitment:* After population suppression, the solutions discarded are randomly generated, which is similar to initialization for maintaining the size of the whole population. Recruitment of new solutions increases the diversity in the population. The solutions are grouped as Immune Network Subset (INS).

(8) *Crossover:* Solutions in the INS will undergo crossover with solutions in CLS with a crossover rate of 0.9. Each solution in the INS will be assigned with another solution in CLS, which are called 'parents'. The crossover rate is the probability of copying a gene (variable) from different parents. For solution from INS, $A = (a_1, a_2, a_3, \ldots, a_n)$; and solution from CLS, $B = (b_1, b_2, b_3, \ldots, b_n)$, the outcome solution (child) $C = (c_1, c_2, c_3, \ldots, c_n)$ can be obtained by:

$$c_i = a_i \quad \text{for } 0 \leq j_i < 0.9 \tag{2}$$
$$c_i = b_i \quad \text{for } 0.9 \leq j_i \leq 1, \tag{3}$$

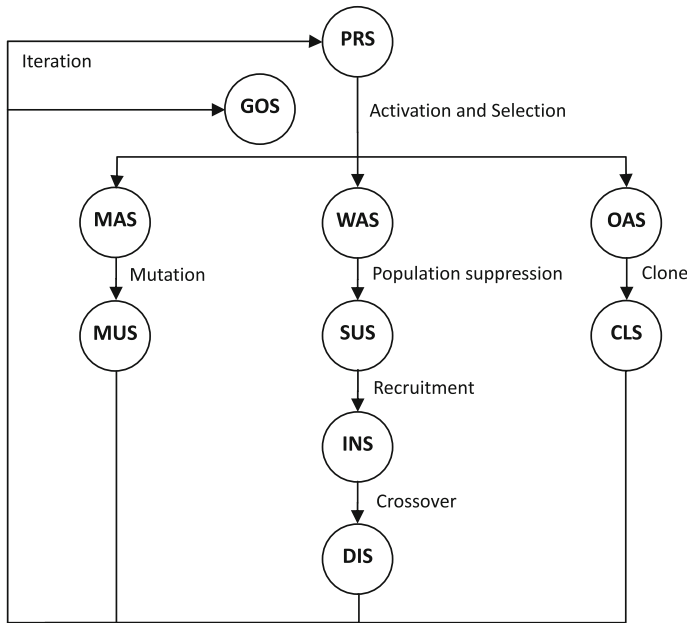where j_i is any random from 0 to 1.

Fig. 1 The IMEA algorithm with its main procedures

A crossover rate of 0.9 means that the probability that a specific gene (variable) of the child copied from its parent A is 0.9 and the probability of copying from parent B is 0.1 (=1−0.9). The crossover rate of 0.9 is chosen because the searching of solutions in the process of crossover will be concentrated in the MAS subset. Solutions after having crossover are grouped as Diversified Subset (DIS).

(9) *Iteration:* The combination of the three resulted sets: CLS, MUS and DIS, will be the new *PRS* for the next simulation. The processes from step 2 to step 8 will be repeated until termination conditions are satisfied. The terminating conditions are set according to the number of iterations, best affinity value or the standard deviation of affinity values in the best subset. The global optimal value is obtained as Global Optimal Solution (GOS) after the termination conditions met.

A number of computer-aided tools are available for the development and simulation of immune-based systems, including C-ImmSim (Baldazzi et al. 2007; Castiglione et al. 2007), IMMSIM (Kleinstein and Seiden 2000), MATLAB, and LISYS (Hofmeyr and Forrest 2000). In this study, a simulation system for the proposed algorithm is developed with MATLAB for its flexibility, dynamic functionality and efficiency in computer simulation (Fig. 1).

4 Algorithm performance benchmarking

Eight single and multi-objective optimization functions are used to benchmark IMEA in solving optimizing problems and obtaining global Pareto fronts (Deb 2001; Man

2006). Unconstrained single objective functions include Sphere Function, Rosenbrock Function, and Step Function (Montiel et al. 2007; Jong and Alan 1975), and unconstrained multi-objective test functions, namely ZDT1, ZDT2, ZDT3, and ZDT4 are used (Zitzler et al. 2000). Constrained multi-objective benchmarking tests are also performed using Constr-Ex (Deb 2001) (Table 2).

4.1 Single objective functions

In this study, IMEA is used to solve the sphere function, Rosenbrock Function, and Step Function to obtain global optimal solutions. The Sphere Function is quadratic, continuous, convex, and unimodal where an optimal solution is obtained by IMEA within 250 iterations (Fig. 2). Rosenbrock function is quadratic, continuous, non-convex, and unimodal where the optimal solution is obtained after 200 iterations (Fig. 3). IMEA is also able to obtain the optimal front for the Step Function, which is discontinuous, non-convex, and unimodal single objective function. The optimal solution obtained is −20 after 160 iterations with the standard deviation decreases rapidly and approaches 1 (Fig. 4).

4.2 Multi-objective functions

Benchmarking with multi-objective functions, namely, ZDT1, ZDT2, ZDT3, ZDT4, and Constr-Ex were also carried out. The algorithm was run for 250 generations with over 30 trials, each started with a population size of 100. Optimal fronts were obtained as shown in Figs. 5, 6 and 7. The solutions obtained from the algorithm reach the Pareto front. It could be seen in the results of ZDT1 and ZDT2 that the solutions are distributed evenly along the Pareto front. For the functions of ZDT3, ZDT4, and Constr-Ex, it is also showed that the solutions reach the optimal front but concentrate in regions. The diversity of the optimal searching capability of the algorithm could be further improved by refining the parameter settings when conducting functions similar to ZDT3, ZDT4, and Constr-Ex.

4.3 Comparison with other evolutionary algorithms

In the performance assessment on an algorithm, reliable assessment methodologies are crucial for validating the algorithm as they involve optimization problems with several solutions and stochastic nature of evolutionary algorithms. Through the assessment, the consistency of the algorithm in approximating and converging to the global Pareto front is measured. The behavior of an evolutionary algorithm during the evolutionary process, including its capabilities to keep diversity and to progressively converge to a set of solutions close to the global Pareto front of a problem, can be analyzed as well throughout the evaluation.

With the goals of multi-objective optimization in converging to the Pareto optimal set as well as maintaining the diversity in solutions of the Pareto optimal set, two performance metrics, Generational Distance (GD) and Spread Metric, are used

Table 2 The benchmarking functions used in the numerical computation

| Test functions	Objective functions	No. of variables	Bound	Global optimal
Sphere function	Minimize : $f(x) = \sum_{i=1}^{n} x_i^2$	30	$-10 \leq x_i \leq 10$	$x_i = 0$
Rosenbrock function	Minimize : $f(x) = \sum_{i=1}^{n-1} [100 * (x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	5	$-2.048 \leq x_i \leq 2.048$	$x_i = 1$
Step function	$f(x) = \sum_{i=1}^{n} x_i$	4	$-5.12 \leq x_i \leq 5.12$	$x_i = -20$
ZDT1	$f_1(x) = x_1; f_2(x) = g(x)\left[1 - \sqrt{x_1/g(x)}\right]$ $g(x) = 1 + 9\left(\sum_{i=2}^{n} x_i\right)/(n-1)$	30	$0 \leq x_i \leq 1$	Convex
ZDT2	$f_1(x) = x_1; f_2(x) = g(x)\left[1 - (x_1/g(x))^2\right]$ $g(x) = 1 + 9\left(\sum_{i=2}^{n} x_i\right)/(n-1)$	30	$0 \leq x_i \leq 1$	Non-convex
ZDT3	$f_1(x) = x_1$ $f_2(x) = g(x)\left[1 - \sqrt{x_1/g(x)} - (x_1/g(x))\sin(10\pi x_1)\right]$ $g(x) = 1 + 9\left(\sum_{i=2}^{n} x_i\right)/(n-1)$	30	$0 \leq x_i \leq 1$	Convex, disconnected
ZDT4	$f_1(x) = x_1; f_2(x) = g(x)\left[1 - \sqrt{x_1/g(x)}\right]$ $g(x) = 1 + 10(n-1) + \sum_{i=2}^{n}\left(x_i^2 - 10\cos(4\pi x_i)\right)$	10	$x_1 \in [0, 1]$ $x_2 \ldots n \in [-5, 5]$	Non-convex
Constraint-Ex	$f_1(x) = x_1$ $f_2(x) = (1 + x_2)/x_1$ subject to $g_1(x) = x_2 + 9x_1 \geq 6$ $g_2(x) = -x_2 + 9x_1 \geq 1$ $0.1 \leq x_1 \leq 1$ $0 \leq x_2 \leq 5$	2	$0.1 \leq x_1 \leq 1$ $0 \leq x_2 \leq 5$	For: $0.39 \leq x_1^* \leq 0.67$ $x_2^* = 6 - 9x_1^*$ For: $0.67 \leq x_1^* \leq 1.00$ $x_2^* = 0$
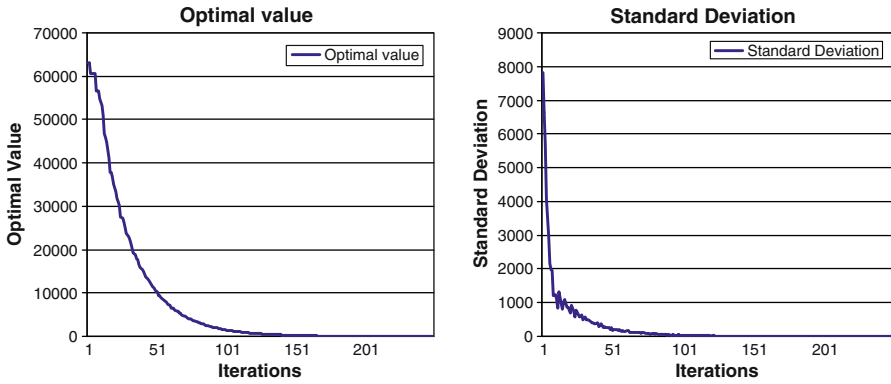
Fig. 2 The optimal front and the standard deviation of the sphere function
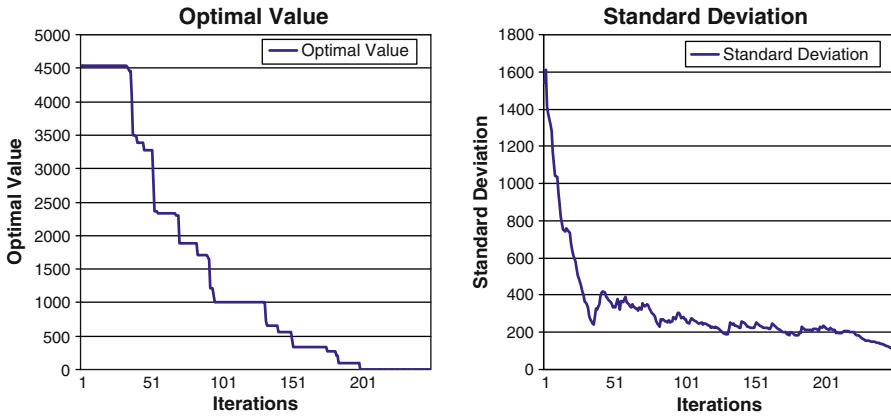

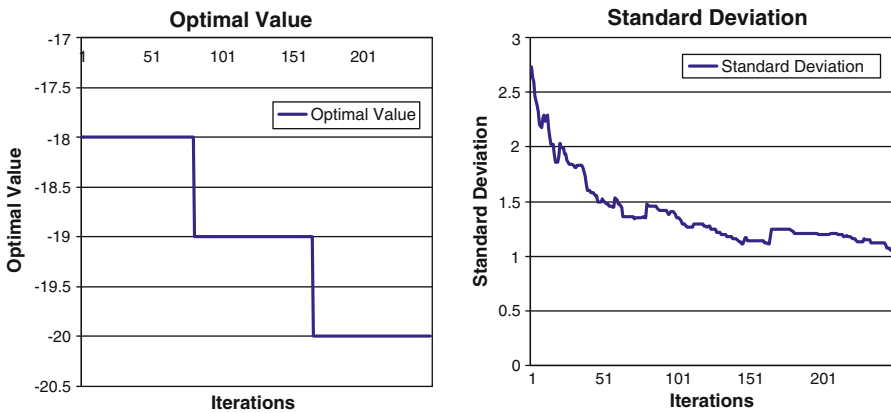**Fig. 3** The optimal front and the standard deviation of Rosenbrock function

Fig. 4 The optimal front and the standard deviation of the step function
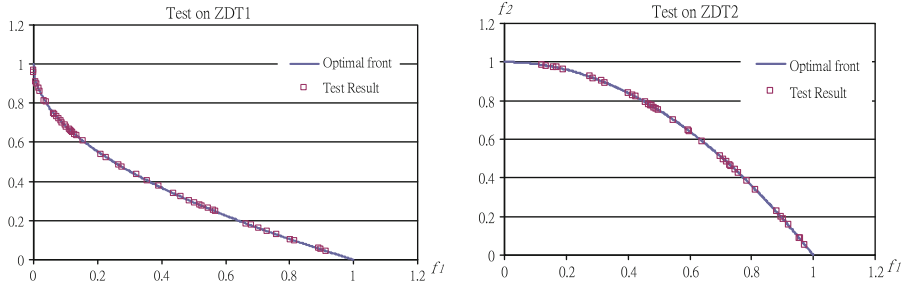
Fig. 5 The optimal fronts of the ZDT1 (*left*) and ZDT2 functions (*right*)
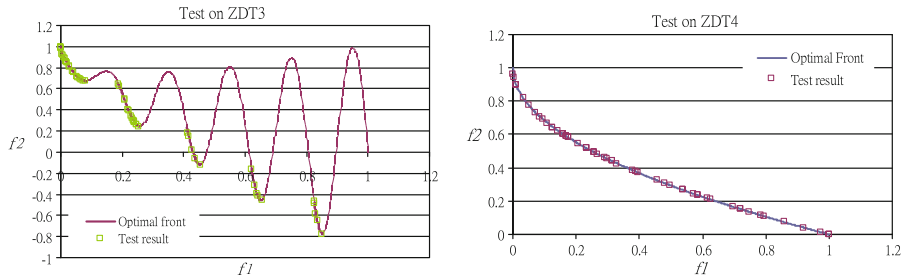
Fig. 6 The optimal fronts of the ZDT3 (*left*) and ZDT4 (*right*) functions
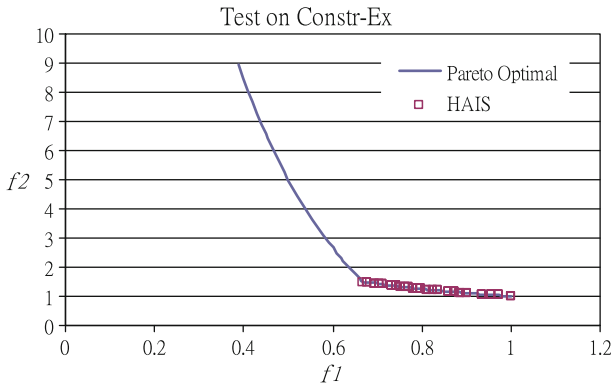
Fig. 7 The optimal front of Constr-Ex function

in the performance assessment. The first metric measures the extent of convergence to a known set of Pareto optimal solutions. The distance between a set of obtained uniformly spaced solutions and their minimum Euclidean distance against the true Pareto optimal front is measured. The average of these distances is calculated as the first metric. The smaller the value of this metric, the better the convergence towards the Pareto-optimal front. As multiple runs are performed, the mean and

variance of the metric are calculated and presented. The second metric measures the extent of spread achieved among the obtained solutions. The Euclidean distances between consecutive solutions in the obtained set of non-dominated solutions are calculated. The Spread Metric is then manipulated using the metric formula to calculate the non-uniformity in the distribution. Details are to be described later in this section. The mean and variance of the Spread Metric are also obtained after multiple runs.

The proposed immunity-based algorithm was compared with a number of multi-objective evolutionary algorithms. Four GA-based algorithms, namely Evolutionary Multi-objective Crowding Algorithm (EMOCA) (Rajagopalan et al. 2005), Non-dominated Sorting Genetic Algorithm-II (NSGA-II) (Deb et al. 2000), Strength Pareto Evolutionary Algorithm II (SPEA-II) (Zitzler et al. 2001; Zitzler and Thiele 1998), and Pareto Achieved Evolution Strategy (PAES) (Knowles and Corne 2000) were compared. The algorithms were executed with the same population size of 100 and iterated for 250 times over 30 trials. The crossover probability (crossover rate) was set to 0.9 and mutation probability (mutation rate) was set to $1/n$ where n is the number of decision variables. Two performance metrics, namely, Generation Distance (GD) and Spread Metric, are introduced for performance comparison.

4.3.1 Generation distance

GD is used to measure the distance between known Pareto front and the true Pareto optimal (Veldhuizen and Lamont 2000).

$$GD = \sqrt{\sum_{i=1}^{|Q|} d_i^2} \tag{4}$$

$|Q|$ refers to the set of solutions generated by the algorithms. The parameter d_i is the Euclidean distance between the solution $i \in Q$ and the nearest solution of true Pareto optimal $P*$ in the objective space. Rajagopalan et al. (2005) suggested that since the Pareto-optimal solutions for the test functions are known, a set of 500 uniformly spaced solutions for each testing function is obtained for calculating the minimum Euclidean distance between the solutions and the Pareto-front. The mean and variance of GD values obtained by the five algorithms are shown in Table 3. IMEA shows a better convergence property than the NSGA-II. In ZDT1 and ZDT 4, IMEA shows good convergence and outweighs all the other algorithms except EMOCA. In ZDT2 and ZDT3, IMEA ranks the fourth among the algorithms. Compared with the well known evolutionary algorithms, IMEA not only shows a similar performance but also shows good convergence property compared to these algorithms. IMEA also demonstrates its competence on multi-objective optimization. It also shows that IMEA is suitable in solving problems having a convex Pareto front with large number of variables to be optimized as shown in ZDT1 and problems with multi-modality property as shown in ZDT4. Overall, IMEA exhibits consistent performance and satisfactory convergence in solving multi-objective functions.

Table 3 The mean and variance of generation distance

Algorithm	Problem	ZDT1	ZDT2	ZDT3	ZDT4
IMEA	Mean	0.0317	0.048	0.0456	0.0499
	Variance	0.0012	0.0003	0.0003	0.0007
EMOCA	Mean	0.029	0.016	0.039	0.022
	Variance	0.0082	0	0.0004	0.0001
NSGA-II	Mean	0.034	0.075	0.11	0.523
	Variance	0.0055	0.00001	0.0006	0.001
SPEA-II	Mean	0.0432	0.033	0.04	0.15
	Variance	0	0.00001	0	0.0015
PAES	Mean	0.032	0.029	0.04	0.338
	Variance	0.00001	0.00001	0	0.00001

Table 4 The mean and variance of spread metric

Algorithm	Problem	ZDT1	ZDT2	ZDT3	ZDT4
IMEA	Mean	0.6276	0.7163	0.7606	0.884
	Variance	0.0028	0.0069	0.0059	0.0071
EMOCA	Mean	0.4024	0.2482	0.4853	0.3072
	Variance	0.0047	0.0023	0.00169	0.0086
NSGA-II	Mean	0.4109	0.4476	0.6898	0.7451
	Variance	0.0018	0.0052	0.0096	0.00178
SPEA-II	Mean	0.6404	0.6437	0.7282	0.195
	Variance	0.014	0.019	0.034	0.113
PAES	Mean	0.531	0.581	0.634	0.412
	Variance	0.038	0.066	0.037	0.07

4.3.2 Spread metric

The spread metric measures the diversity of the solutions (Deb et al. 2000):

$$\Delta = \frac{\sum_{m=1}^{M} d_m^e + \sum_{i=1}^{|Q|} |d_i - \overline{d}|}{\sum_{m=1}^{M} d_m^e + |Q|\overline{d}} \tag{5}$$

The parameter d_i is the Euclidean distance between adjacent solutions and d_m^e is the Euclidean distance between extreme solutions of the Pareto optimal front and the nearest solutions of Q corresponding to the mth objective. A smaller value of Δ represents a more uniformly spaced set of solutions. The mean and variance of the spread metric over 30 trials are shown in Table 4. IMEA shows a slightly higher mean and lower variance compared with other algorithms. IMEA outperformed SPEA-II and PAES in variance. Overall, IMEA performs a satisfactory on diversity of the solutions generated

but it shows there are rooms for improvements on performing a more diversified search on the global optimal solution.

In general, it can be observed from Sect. 4.2 that IMEA are able of finding solutions near the global Pareto front and it also demonstrates a good convergence capability in optimization problems. It also shows that IMEA are suitable to solve problems having a convex Pareto front with large number of variables and functions with multi-modality property. These properties are suitable for applying the developed algorithm in solving industrial problems in container repositioning problems which exhibit these characteristics.

5 Container repositioning case study

In container transportation and logistics industry, container traffic and trade have been continuously growing. The container traffic has increased from 28.7 million twenty-foot equivalent unit (TEU) in 1990 to 152 million TEU in 2008, an increase of about 430%. This corresponds to an average annual compound growth of 9.5%. This corresponds to an average annual compound growth of 9.5%. The compound annual growth rate for global container trade volumes from 2005 to 2015 is estimated to be 7.6%. The increase in the volume of container traffic due to the overall growth of global trade, popularization of the use of containerized goods, deployment of mega vessels, and increase in trade imbalance have induced various complex issues in liner shipping. Large volume of container traffic leads to the concern of empty container repositioning. It has been indicated in many research studies that empty container repositioning has been accounted for about 20% of all ocean container movements (Mongelluzzo 2004; United Nations 2007). As the annual global container traffic increases steadily, the costs in empty container movements of ship liners will also be increased, incurring millions of US dollars each month, subject to the container trade volume. Investigations on effective and efficient empty container repositioning strategies have raised the attention of liners in recent years.

The developed evolutionary algorithm is implemented in an international container transportation and logistics corporation to solve the container repositioning operations. Due to trade imbalance, empty containers are repositioned from surplus locations to deficit locations to supply the large demand for containers used in deficit regions. The algorithm has been applied to tackle the repositioning from Europe to Asia Pacific regions, from U.S. to Asia Pacific regions, and Australia to Asia Pacific regions. The developed algorithm assists to improve the decision making on the quantity of repositioning and minimizes the transportation costs incurred during the shipment operations covering weekly repositioning costs of over USD millions. Below sections presented the problem definition, selected real industrial cases, optimization results, analysis, and discussions of the container repositioning case study.

5.1 Problem definition

The multi-objective container repositioning optimization model assumes that the empty container repositioning operation follows a fixed regular service route pattern

determined by the liner operator. It is further assumed that demand from customers is always met and shipping requirement cannot be delayed to the next time period. The surplus and deficit locations being studied are considered in the direct call port of the vessel services. In the optimization process, it is also assumed that each unit of container is assumed to occupy one twenty-foot equivalent unit (TEU). The objective functions for the total repositioning cost and total dissatisfied demand are given by:

$$\text{Minimize } f_1 : C_{\text{Total}} = \sum_{i=1}^{|S|} \sum_{j=1}^{|D|} \sum_{k=1}^{|V|} \left(C_{ijk} Q_{ijk} \right) \tag{6}$$

$$\text{Minimize } f_2 : U_{\text{Total}} = \sum_{j=1}^{|D|} U_j \tag{7}$$

where C_{ijk} is the repositioning cost of shipping an empty container from a surplus location $i \in S$ to a deficit location $j \in D$ with vessel service $k \in V$; Q_{ijk} is the planning quantity of empty containers shipping from a surplus location i to deficit location j through vessel service k; and U_j is the unit of dissatisfied demand in deficit location j.

The objective functions are subject to the following constraints:

(1) *Supply constraints*: the total repositioning quantity from the ith surplus location must not exceed the maximum supply units of the ith surplus location Q_i

$$\sum_{j=1}^{|D|} \sum_{k=1}^{|V|} Q_{ijk} \le Q_i \quad \text{for } i = 1, 2, \ldots, |S| \tag{8}$$

(2) *Vessel space constraints*: the total repositioning quantities by the kth vessel service must not exceed the maximum available space of the kth vessel service Q_k

$$\sum_{i=1}^{|S|} \sum_{j=1}^{|D|} Q_{ijk} \le Q_k \quad \text{for } i = 1, 2, \ldots, |V| \tag{9}$$

(3) Integer and non-negativity constraints on repositioning quantities:

$$Q_{ijk} \ge 0 \quad \text{and} \quad Q_{ijk} \in I \tag{10}$$

5.2 Simulation experiments

A number of simulation studies were performed based on the repositioning operations between Europe and Asia Pacific regions with eight selected surplus locations, twelve deficit locations, eight service routes involving 112 decision variables, and hypothetical repositioning costs of respective repositioning routing combinations.
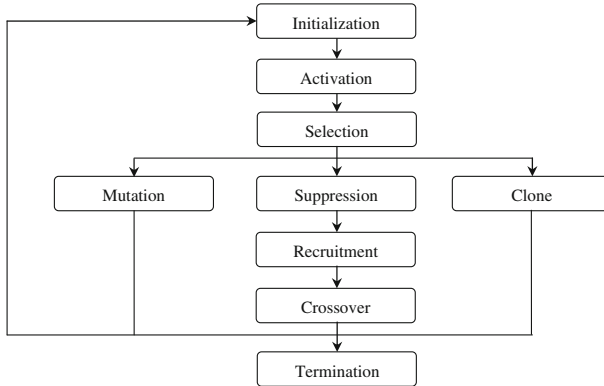
Fig. 8 Operation of IMEA in the case study

A representation matrix is supplemented in Appendix II to show the background of the total number of decision variables, the corresponding vessel service structure, number of surplus and deficit locations, and the repositioning unit costs of the corresponding repositioning port pairs in this study. Based on the results of benchmarking with multi-objective optimization functions, the repositioning cases are solved by implementing the algorithm with MATLAB as described in Sect. 3.2 with the parameter settings of IMEA given in Sect. 4. These parameters are set based on the desired flexibility, dynamic functionality, and efficiency in computer simulation. The steps of the algorithm (Sect. 3.2) are applied to the problem of optimization of container repositioning as follows:

(1) *Initialization*: random sets of repositioning quantities are generated as initial population.
(2) *Activation*: affinity values of the solution sets are manipulated based on the objective function defined.
(3) *Selection*: sets of repositioning quantities are sorted with their corresponding affinity values based on the primary and secondary criterion.
(4) *Cloning*: the optimal sets of solutions in the population are cloned and stored for next iteration.
(5) *Mutation*: selected sets of solutions in the population undergo mutation.
(6) *Suppression*: the remaining sets of solutions in the population, which are the sub-optimal sets of solutions with the small affinity values and large crowded distances, are discarded.
(7) *Recruitment*: the discarded set of solutions are replaced with randomly generated new solution sets to increase the diversity of the population.
(8) *Crossover*: the new sets of randomly generated solutions undergo crossover with the cloned solutions.

The pseudo-codes of the above major steps are shown in Appendix 1 and the logic flow is described in Fig. 8.

Table 5 Demand and supply quantities for Case 1

Demand locations	Quantity	Supply locations	Quantity
Tianjin	200	Amsterdam	520
Dalian	150	Hamburg	1,300
Qingdao	180	Southampton	1,740
Ningbo	450	Le Havre	285
Pusan	88	Antwerp	828
Xiamen	160	Rotterdam	1,228
Hong Kong	350	Genoa	190
Yantian	450	Fos	75
Shanghai	750	Total	6,166
Shekou	1,300		
Singapore	700		
Kaohsiung	100		
Total	4,878		
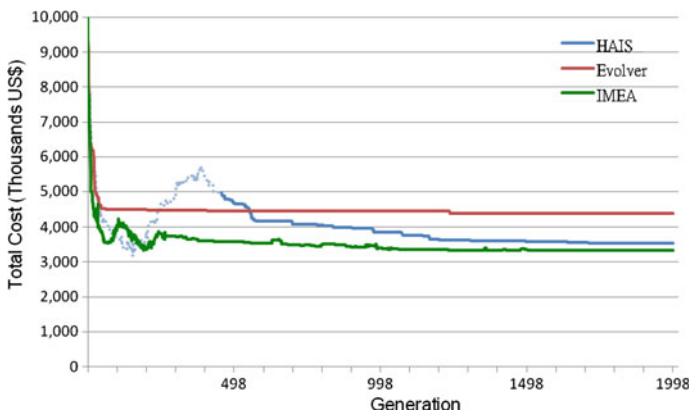
Fig. 9 Simulation results of single-objective container repositioning problem

5.3 Simulation results

Five of the simulated cases are illustrated below. The demand and supply quantities vary in each case with an example of the quantities in Case 1 shown in Table 5 having demand quantities of 4,878 units and supply quantities of 6,166 units. The simulations are carried out with a population size of 50 and 1,500 iterations for each trial. The simulation result of a single-objective container repositioning considering only costs as define by Eq. (6) is plotted against number of generation is shown in Fig. 9. A comparison with two other algorithms, namely Evolver (Laguna 1997) and HAIS (Wong et al. 2008) were conducted. Evolver is a GA-based algorithm used to solve single-objective optimization problem. HAIS is an AIS-based hybrid algorithm, an

Table 6 Repositioning simulation results obtained by using IMEA and current practices

Case	IMEA		Current Practice	
	Repositioning Rcost (USD)	Dissatisfaction level (%)	Repositioning cost (USD)	Dissatisfaction level (%)
1	800,709	22.60	1,167,570	21.63
2	839,550	21.38	1,217,570	20.67
3	826,792	22.75	1,209,130	22.29
4	751,008	20.34	1,207,110	21.31
5	726,110	21.92	1,193,620	22.42
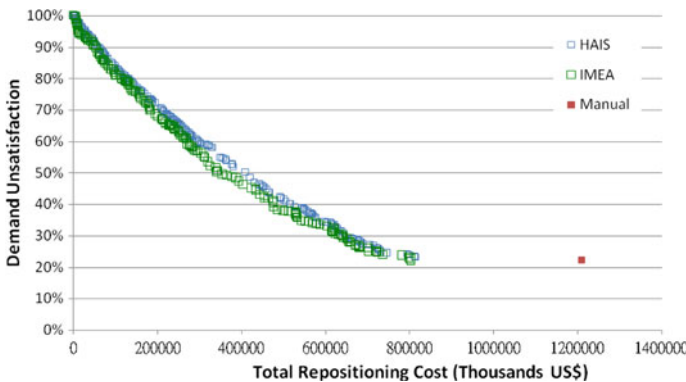
Fig. 10 Simulation result of multi-objective container repositioning problem (Case 1)

earlier research conducted before IMEA. IMEA further improves the exploration process with mutation, memory archive, immuno-suppression, crossover, and recruitment carried out in a parallel fashion. Subsets with roles as operators in the algorithm are added in IMEA to carry out various functions. The evaluation and selection of the solution population are also added with the Harmonic average distance to evaluate the crowding degree of the solution. In IMEA, it is proposed to be developed using MATLAB as the programming platform. The multi-objective optimization problem with repositioning quantities using IMEA fulfils the empty container demand with regards to costs and the percentage of fulfillment failure is further simulated. The result is also compared with the quantities obtained by current industry practices of a typical global shipping line as shown in Table 6. The simulated result of Case 1 is shown in Fig. 10 with a comparison of the results from HAIS and the current practices in manipulating the results. It shows a more optimal solution than the current practices regarding the repositioning costs and dissatisfaction level. It also illustrated that IMEA has a better rate of convergence as compared with HAIS. The differences in costs shown from the results of IMEA and the current practice illustrated that there is a cost saving in the routing with their corresponding proposed repositioning port

pairs. Although the dissatisfaction level of both methods are in the range of one-fifth of the demand, considering the overall supply from Europe and Intra-Asia Pacific, this percentage can be minimized by the arrangement of repositioning within the Asia Pacific regions.

From the results, IMEA is shown to be a viable optimization algorithm and as a decision support tool in assisting shipping liners to make decision and plan for global repositioning of empty containers in a cost effective manner. The capability of IMEA on performing global search for the Pareto front has also been demonstrated in the single and multiple objective benchmarking exercises in Sect. 4. These findings show that IMEA is a practical and useful optimization tools.

It is envisaged that further improvements could be made on the spread of its search capability discussed in Sect. 4.1 by exploring different approaches to suppression and crossover processes (Sect. 3). Exploration of crossover methodology may also assist the rate of convergence, thus resulting in a more efficient global optimal searching process. Further enhancement on the system could be made to allow different container sizes and types to be considered.

6 Conclusion

This paper contributes the exploration of potential AIS theories, development of novel immunity-based evolutionary optimization algorithm, and application of the developed algorithm on global maritime logistics industrial optimization problems. A novel immunity-based evolutionary algorithm known as Immunity-based Evolutionary Algorithm (IMEA) is developed for solving multi-objective optimization problem such as those found in global repositioning of empty containers. The algorithm integrates the distinct characteristics of Clonal selection and immune suppression of artificial immune systems, and genetic theories, with a sorting scheme featuring uniform crossover, multi-point mutation, non-dominance and average harmonic distance sorting to attain the Pareto optimal front in an efficient manner. The algorithm has been studied using benchmarking functions and comparison with other similar algorithms, including EMOCA, NSGA-II, SPEA-II, and PAES. In this paper, the algorithm is implemented with MATLAB to solve a practical container repositioning problem where promising simulation results were obtained. Various cases of container repositioning from selected surplus locations in Europe to deficit locations in Asia Pacific regions through available service routes involving a large number of decision variables are studied. The developed algorithm demonstrates itself as a useful intelligent decision support tool providing an optimal solution based on a combination of the best empty container demand fulfillment and minimum repositioning costs. As such, the algorithm has the potential of solving complex multi-objective optimization problems such as container repositioning operations in global liner shipping business in a timely and cost effective manner.

Appendix I: Matlab coding on IMEA for multi-objective optimization

1.1 Activation

```
A = PRS;
[Q,R] = qr(A);
[Q1,R1] = qrinsert(Q,R,i+1,Nondominated_sort);
A = Q1*R1;
[Q,R] = qr(A);
[Q1,R1] = qrinsert(Q,R,i+2,Affinity_value3);
z = Q1*R1;
A = sortrows(z,[i+1 i+2]);
[Q,R] = qr(A);
[Q1,R1] = qrdelete(Q,R,i+2);
A = Q1*R1;
[Q,R] = qr(A);
[Q1,R1] = qrdelete(Q,R,i+1);
z = Q1*R1;
OAS = z(1:round(0.3*j),:);
CLS = OAS;
MAS = z(round(0.3*j)+1:round(0.7*j),:);
WAS = [z(round(0.7*j)+1:round(0.9*j),:);randsol(j-round(0.9*j))]
```

1.2 Selection

```
n = zeros(b,1);
for count = 1:b-1
    c = count;
    for count = c+1:b
      k = count;
        if Affinity_value1(c)>Affinity_value1(k) & Affinity_value2(c)>Affinity_value2(k)
              n(c) = n(c) + 1;
        elseif Affinity_value3(c)>Affinity_value3(k) & Affinity_value2(c)>Affinity_value2(k)
              n(c) = n(c) + 1;
         elseif Affinity_value1(c)>Affinity_value1(k) & Affinity_value3(c)>Affinity_value3(k)
              n(c) = n(c) + 1;
              else
                  n(c) = n(c);
            end;
    end;
end;
Nondominated_sort = n;
```

1.3 Clone

```
z = Q1*R1;
OAS = z(1:round(0.3*j),:);
CLS = OAS;
```

1.4 Suppression

```
For count = 1:b;
    k = count;
    for count = 1:b;
        m = count;
        e                                                                    =
(Affinity_value1(k)-Affinity_value1(m)).^2+(Affinity_value2(k)-Affinity_value2(m)).^2;
        c(k,m) = sqrt(e);
    end;
end;
end;

e = 2;
f = sort(c,2);
p = f(:,2:166);
g = zeros(b,1);

for count = 1:b;
    n = count;
    h = p(n,2:5);
    i = sum(1./h);
    g(n) = 4/i;
end;
```

1.5 Mutation

```
f = ceil(rand*112);
if f==1
        g = [(0.8+(rand*0.2))*MAS(1,1) MAS(1,2:c)];
else if f==112
    g = [MAS(1,1:c-1) (0.8+(rand*0.2))*MAS(1,c)];
else
    g = [MAS(1,1:f-1) (0.8+(rand*0.2))*MAS(1,f) MAS(1,f+1:c)];
end;
end;

h = g;

for count = 2:b;
    k = count;
    f = ceil(rand*112);

    if f==1
            g = [(0.8+(rand*0.2))*MAS(k,1) MAS(k,2:c)];
    else if f==112
            g = [MAS(k,1:c-1) (0.8+(rand*0.2))*MAS(k,c)];
        else
            g = [MAS(k,1:f-1) (0.8+(rand*0.2))*MAS(k,f) MAS(k,f+1:c)];
    end;
        end;

    i = [h; g];
    h = i;

end;
```

1.6 Recruitment

```
function PRS = randsol(j);

global PRS Objective_function
a = size(Objective_function);
b = a(1);
c = round(1000*rand(j,b));
d = zeros(j,b);

A = [520 ;1300 ;1780 ;250 ;828 ;1228 ;160 ;140];
for count = 1:j;
k = count;
```

1.7 Crossover

```
For count = 1:c;
    k = count;
        for count = 1:d;
            m = count;
            n = ceil(rand*e);
            if rand > 0.9
                WAS(k,m) = CLS(n,m);
            else
                WAS(k,m) = WAS(k,m);
            end;
        end;
end;

DIS = WAS;
```

Appendix II: Repositioning demand and supply quantities and costs

Repositioning Quantity on supplying Asia-Pacific Regions from European Regions

Asia-Europe Trade Services																			
Services		AE1		AE2			AE3			AE 4			AEM		NW1		NW2	NW3	Total
Demand	QTY	AMS	HAM	LEH	AMS	HAM	ANR	STN	HAM	RTM	RTM	HAM	STN	GOA	FOS	HAM	LEH	ANR	HAM	Supply
ZIN	50										0	37	13							50
DAN	15										0	3	12							15
QIN	20										0	15	5							20
NIN	40							7	0	0	0	2	1	25	6					40
PUS	15										0	10	5							15
XIA	40							26	2	0								12		40
HKG	80			0	3	0	0	69	0	0				5	2	0	0	0	0	80
YAT	90			80	0	0	0	10	0	0								0		90
SHA	90							6	0	0	0	0	26	48	10			0		90
SKZ	80			0	17	0	30	10	0	0				22	2					80
SIN	50	50	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	50
KHH	10			0	0	0	0				0	0	10			0	0			10
Total	580	50	0	80	20	0	30	128	2	0	0	66	72	100	20	0	0	12	0	580

Simulated repositioning cost from Europe to Asia Pacific regions

		AE1		AE2				AE3			AE 4			AEM		NW1		NW2	NW3	Total
Services		AMS	HAM	LEH	AMS	HAM	ANR	STN	HAM	RTM	RTM	HAM	STN	GOA	FOS	HAM	LEH	ANR	HAM	
Demand	QTY																			$upply
ZIN	50										232	216	163							611
DAN	20										239	223	170							632
QIN	20										245	229	176							650
NIN	60							172	225	241	241	225	172	138	216					1630
PUS	20										316	300	247							863
XIA	50							165	218	234								173		790
HKG	80			341	209	299	256	245	299	314				211	289	299	341	256	299	3658
YAT	100			250	225	315	272	261	315	331								272		2241
SHA	90							168	222	237	237	222	168	134	212			178		1778
SKZ	80			240	168	258	214	204	258	273				170	248					2033
SIN	50	142	232	274	142	232	189	179	232	248	248	258	179	145	223	232	274	189	232	3850
KHH	20			230	98	188	145				204	188	134			188	230			1605
Total	640	142	232	1335	842	1292	1076	1394	1769	1878	1962	1861	1409	798	1188	719	845	1068	531	20341

References

Baldazzi V, Castiglione F, Bernaschi M (2007) An enhanced agent based model of the immune system response. Cellular Immunology, vol. 244, Issue 2, December 2006, pp 77–79. International Conference on Immunogenomics and Immunomics, Budapest, Hungary, October 8–12, 2006

Burnet FM (1957) A modification of Jerne's theory of antibody production using the concept of clonal selection. Aust J Sci 20:67–69

Castiglione F, Duca K, Jarrah A, Laubenbacher R, Hochberg D, Thorley-Lawson D (2007) Simulating Epstein–Barr virus infection with C-ImmSim. Bioinformatics 23(11):1371–1377

Chen J, Mahfouf M (2006) A population adaptive based immune algorithm for solving multiobjective optimization problems. In: Bersini H, Carneiro J (eds). ICARIS 2006. LNCS, vol 4163, pp 280–293

Coello Coello CA, Cortes NC (2002) An approach to solve multiobjective optimization problems based on an artificial immune system. Artificial Immune Systems: First International Conference, ICARIS 2002

Coello Coello CA, Rivera DC, Cortes NC (2003) Use of an artificial immune system for job shop scheduling. Artificial immune systems: Proceedings of the second international conference, ICARIS 2003, Edinburgh, UK, September 1–3. Springer, Berlin, pp 1–10

Craninic TG, Gendreau M, Dejax P (1993) Dynamic and stochastic models for the allocation of empty containers. Oper Res 41(1):102–126

Dantzig GB, Ramser JH (1959) The truck dispatching problem. Manage Sci 6(1):80–91

de Castro LN, Von Zuben FJ (2000) The clonal selection algorithm with engineering applications. In: Workshop Proceedings of GECCO'00, pp 36–37, Workshop on artificial immune systems and their applications, Las Vegas, USA

Deb K, Pratap A, Agarwal S, Meyarivan T (2000) A fast and elitist multi-objective genetic algorithm: NSGA-II. In: Proceedings of the parallel problem solving from nature V1, pp 849–858

Deb K (2001) Multi-objective optimization, multi-objective optimization using evolutionary algorithms. Wiley, New York, pp 13–48

Feng CM, Chang CH (2008) Empty container reposition planning for intra-asia liner shipping. Maritime Policy Manage 35:469–489

Finke G, Claus A, Gunn E (1984) A two-commodity network flow approach to the traveling salesman problem. Congress Numerantium 41:167

Florez H (1986) Empty container repositioning and leasing: an optimization model. Ph.D. Dissertation, Polytechnic Institute of New York

Fonseca CM, Fleming PJ (1995) An overview of evolutionary algorithms in multiobjective optimization. Evol Comput 3(1):1–16

Garain U, Chakraborty MP, Dasgupta D (2006) Recognition of handwritten indic script using clonal selection algorithm. In: 5th international conference (ICARIS). Lecture notes in computer science. Oeiras, Portugal, pp 256–266

Golden BL, Raghavan S, Wasil EA (2008) The vehicle routing problem: latest advances and new challenges. Springer, New York

Guzella TS, Mota-Santos TA, Caminhas WM (2007) A novel immune inspired approach to fault detection. In: ICARIS 2007, pp 107–118

Hofmeyr SA, Forrest S (2000) Architecture for an artificial immune system. Evol Comput 8(4):443–473

Huang VL, Suganthan PN, Qin AK, Baskar S (2006) Multi-objective differential evolution with external archive and harmonic distance-based diversity measure, Nanyang Technological University, Singapore, Tech. Rep. TR-07-01

Jong D, Alan K (1975) Analysis of the behavior of a class of genetic adaptive systems, Engineering, College of-Technical Reports, University of Michigan, 1975

Jozefowiez N, Semet F, Talbi E (2006) Enhancements of NSGA II and its application to the vehicle routing problem with route balancing. Lect Notes Comput Sci 3871:1611–3349

Keko H, Skok M, Skrlec D (2003) Artificial immune systems in solving routing problems, EUROCON 2003. Computer as a Tool. The IEEE Region 8, vol 1, pp 62–66

Kleinstein SH, Seiden PE (2000) Simulating the immune system. Comput Sci Eng 2(4):69–77

Knowles JD, Corne DW (2000) Approximating the non-dominated front using the Pareto archived evolution strategy. Evol Comput 8(2):149–172

Laguna M (1997) Metaheuristic Optimization with Evolver, Genocop and OptQuest. Available online via http://www.crystalball.com/optquest/paperlist.html

Luh GC, Wu CY, Cheng WC (2004) Artificial immune regulation (AIR) for model-based fault diagnosis, Artificial Immune Systems: Third International Conference, ICARIS 2004

Man MR (2006) Single- and multiple-objective optimization with differential evolution and neural networks. VKI lecture series: introduction to optimization and multidisciplinary design, March 6–10, 2006

Mongelluzzo B (2004) Thinking inside the box. J Commerce, p 1

Montiel O, Castillo O, Melin P, Diaz AR, Sepulveda R (2007) Human evolutionary model: a new approach to optimization. Inf Sci 177:6–9

Nicosia G, Castiglione F, Motta S (2001) Pattern recognition by primary and secondary response of an artificial immune system. Theory Biosci 120(2):93–106

Nicosia G, Cutello V (2002) An immunological approach to combinatorial optimization problems, Advances in artificial intelligence. IBERAMIA 2002, Proceedings of 8th Ibero-American Conference, November 12–15, 2002. Lecture notes in computer science, vol 2527, pp 361–370

Omkar SN, Khandelwal R, Yathindra S, Naik GN, Gopalakrishnan S (2008) Artificial immune system for multi-objective design optimization of composite structures. Eng Appl Artif Intell 21(8):1416–1429

Rajagopalan R, Mohan CK, Mehrotra KG, Varshney PK (2005) An evolutionary multi-objective crowding algorithm (EMOCA): benchmark test function results. IICAI 2005, pp 1488–1506

Sun M, Wang X, Chen X, Cao L (2009) Study on empty container repositioning problem under sea-rail through transport. Second international conference on intelligent computation technology and automation 3:771–774

Swiecicka A, Seredynski F, Zomaya AY (2006) Multiprocessor scheduling and rescheduling with use of cellular automata and artificial immune system support. IEEE Trans Parallel Distrib Syst 17(3):253–262

Tan KC, Lee TH, Khor EF (2001) Evolutionary algorithms with dynamic population size and local exploration for multiobjective optimization. IEEE Trans Evol Comput 5(64):565–588

Taylor DW, Corne DW (2003) An investigation of the negative selection algorithm for fault detection in refrigeration systems. In: Proceedings of the second international conference, ICARIS 2003, Edinburgh, UK, September 1–3, 2003. Springer, Berlin, pp 34–45

Timmis J, Neal M (2000) Investigating the evolution and stability of a resource limited artificial immune system. In: Proceedings of the genetic and evolutionary computation conference, workshop on artificial immune systems and their applications, pp 40–41

Ting JH, Cheung RK, Chen CY (1996) Stochastic and dynamic network optimization model for minimizing empty container allocation costs. In: Industrial Engineering Research-Conference Proceedings, 1996, pp 275–280, Proceedings of the 1996 5th Industrial Engineering Research Conference, Minneapolis, MN, USA, May 18–20, 1996

United Nations (2007) Regional shipping and port development—container traffic forecast, New York

Veldhuizen DAV, Lamont GB (2000) On Measuring Multiobjective Evolutionary Algorithm Performance. In: Proceedings of the 2000 Congress on Evolutionary Computation 1:204-211

Vrugt JA, Robinson BA (2007) Improved evolutionary optimization from genetically adaptive multimethod search. Proc Natl Acad Sci USA (PNAS) 104(3):708–711

White JA, Garrett SM (2003) Improved pattern recognition with artificial clonal selection? Artificial immune systems: Proceedings of the second international conference, ICARIS 2003, Edinburgh, UK, September 1–3, 2003. Springer, Berlin, pp 181–193

Wong EYC, Yeung HSC, Lau HYK (2008) Immunity-based hybrid evolutionary algorithm for multi-objective optimization in global container repositioning. Eng Appl Artif Intell (in Press)

Zitzler E, Deb K, Thiele L (2000) Comparison of multi-objective evolutionary algorithms: empirical results. Evol Comput J 8(2):125–148

Zitzler E, Laumanns M, Thiele L (2001) SPEA2: Improving the strength pareto evolutionary algorithm, Swiss Federal Institute of Technology, Tech-Rep. TIK-Rep, 103, 2001

Zitzler E, Thiele L (1998) Multiobjective optimization using evolutionary algorithms—a comparative study. In: Proc. Parallel Problem Solving from Nature V, pp 292–301

Zitzler E, Thiele L (1999) Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. IEEE Trans Evol Comput 3(4):257–271