The Development of Hybrid Cross Entropy-Genetic Algorithm for Multi-Product Inventory Ship Routing Problem with Heterogeneous Fleet

Budi Santosa and Rita Damayanti
Department of Industrial Engineering
Surabaya Institute of Technology, Indonesia

Abstract

This research develops multi-product inventory ship routing problem with heterogeneous fleet. ISRP is a problem that combines inventory level management in every unloading port and the routing process of the ship. The problem that developed in this research considers some different things those are the weight of ships that can be moored in some ports, product compatibility, port setup and compartment washing. And the objective function of this problem is minimizing travelling cost, port setup cost, ship charter cost, and compartment washing cost. With the constraint that forbids some ships moored in some ports, there will be process to choose a ship in order to obey the constraint and also make sure that the solution will give minimum cost. Besides choosing the ship, this research also find the best product allocation, the best route for every ships and the best shipping quantity. ISRP is one of NP hard problem. The solution of this problem needs a high computation time regarding to the complexity of the problem. So in this research, it will be developed meta- heuristic method by hybridizing Cross Entropy and Genetic Algorithm. Cross Entropy is chosen because this algorithm can solve NP hard problem well and easy to apply in combinatorial problem. But this algorithm needs a high computation time to create a new sample. So due to this lack, CE will be hybridized with GA in order to get new samples fast by mutation step. This research also performs another method to compare the performance of hybrid CEGA. The solution of CEGA will be evaluated by comparing them to the output that hybrid Tabu Search has. And the outcome shows that CEGA gives better solution but the computation time is longer than hybrid TS**.**

Keywords
Multi-product, inventory ship routing problem, heterogeneous fleet, Cross Entropy-Genetic Algorithm

1. Introduction

Product shipments by sea plays a large role in global trade, it's estimated about 65% to 85% of global trade that utilize delivery by sea [1]. It also stated that the product delivery through this pathway requires the most inexpensive cost when compared to other types of transportation [2]. There are several problems in the product shipment by sea, one of them is inventory ship routing problem (ISRP). ISRP integrates inventory management problems of each port and the determination of the ship [3]. When performs the routing process, the selecting of ships must be done because not of all ports can be visited by them. This is cause by the capacity that owned by each port. The capacity should be adjusted to deadweight tonnage that ships have [4]. Besides that, the product compatibility also needs to be considered when loading the products in the ship [5]. Because not of all products can be placed in one ship. Except that, it also needs to consider the product that the ship loads before. If the product that will be loaded is different from the previous, the compartment of the ship must be washed first [6]. So when performs the loading activity, it needs to consider the washing cost to minimize total cost. Research of the ISRP has been done by several researchers. But there are no researches that consider the ship which can dock at the port, port setup, and compartment washing simultaneously. So in this research, it will be developed multi-product inventory ship routing product with heterogeneous fleet that consider the three things.

Multi-product ISRP with heterogeneous fleet is NP-hard problem. The problem solving requires a long computation time by the increase of problem size. If this problem is solved by the exact method, it needs long computation time to obtain optimal solution. Therefore, in this research will be developed hybrid Cross Entropy-Genetic algorithm that can produce solutions quickly and also avoid the trap of local optimum. CE is selected because it has power to solve NP-hard problem [7]. In addition, CE is easy to be applied in combinatorial problem [8]. CE needs longer computation time if it stands alone. One of the algorithms that can be hybridized with CE to reduce the computing time is Genetic algorithm [9]. With GA, a new sample can be obtained quickly through mutation mechanism. The mutation types that will be used in this research are swap, insertion, inversion, forward and backward mutation. By this mutation, the sample will spread in solution space and the local optima trap can be avoided. To accelerate the computing time, the crossover mechanism in GA will be eliminated because the process is complicated. CEGA performance will be tested by comparing the solution with hybrid Tabu Search method that developed before.

2. Mathematical Model

The problem that developed in this research has limitations, they are the developed model does not pay attention to natural phenomena such as wind, waves, evaporation, tidal and others. In addition, when determining the ship to dock in the harbor, it just considers deadweight tonnage factors while the ship length and draft are not considered. Models also assume that the products are always available at the loading port during the planning horizon that has been determined, the consumption rate at each port and the speed of the ship is constant, the unloading process at the port can be done by more than one ship simultaneously and unloading process of second products and so on at the same port and ship can be done immediately after the previous process is completed without having to wait the operations hour, and the last is the unloading process will continue until completed despite passing the operations hour.

Multi-product inventory ship routing problem in this research can be applied to the distribution of oil by tanker problem. The compartment in this research is undedicated, so ship can carry any type of product, but only one type at one time. Ship that used in this research is a chartered ship with time charters so the system is not influenced by the amount that ship brought, but based on the time and duration of the charter [10]. In one trip, the ship could visit more than one unloading port due to the capacity of the ship is greater than the amount of cargo that must be delivered. And when all the unloading port already has sufficient inventories to meet its needs until the end of the planning horizon, the ship will return to the loading port.

Loading port can receive ships with any tonnage and open for 24 hours. Unloading port consume more than one product with specified consumption rate. Unloading port has a limited operational time (time windows) that called daylight. Both loading and unloading ports can serve multiple ships at the same time. But one ship cannot load and unload different products in the same time. In the model developing, there are some points that will be considered, they are: the weight of the ship that can be docked in every port, the setup time is not equal to zero and do not ignore the washing compartment of the ship. Product compatibility constraint that used in this research is products that are not compatible with each other cannot be load on the same ship. This is done to adjust the real conditions.

The mathematical model in this research can be explained below.

<u>Decision variables</u>

x_{imjnv} : 1 if the ship moves from unloading port (i, m) to the unloading port (j, n).
x_{imdnv} : 1 if the ship moves from unloading port (i, m) to the loading port (d, n).
y_{im} : 1 if position (i, m) is not reachable.
r_{ik} : 1 if the stock level of all unloading port was able to meet the consumption needs.
u_{iv} : 1 if the deadweight tonnage of the ships is smaller than unloading port capacity.
z_c : 1 if compartment is washed before load the next cargo.
by_{dmkvc}: 1 if there is product that loaded in loading port.
xy_{imkv}: 1 if product is brought by the ship in the m arrival.
I_{imvkc} : The amount of products that ship bring after leave unloading port.
q_{imvkc} : The number of products unloaded.
q_{dmvkc} : The number of products loaded.
p_{imv} : 1 if ship visit unloading port and do the port setup.
p_{dmv}: 1 if ship visit loading port and do the port setup.
ta_{im} : Starting processing time in unloading port.
ta_{dm} : The arrival time of the ship at loading port.
te_{im} : Ending Service time in unloading port.
te_{dm} : Ending Service time in loading port.
s_{imk} : Product stock levels at unloading port.

Mathematical model

Objective function:

Minimize

$$\sum_{v\in V}\sum_{(i,m,j,n)\in A_v} c_{ijv}x_{imjnv}+\sum_{v\in V}\sum_{(d,m,j,n)\in A_v} c_{djv}(x_{dmjnv}+x_{imdnv})+\sum_{v\in V} cp_{iv}\left(\sum_{(i,m)\in N} p_{imv}+\sum_{(d,m)\in U} p_{dmv}\right)+\sum_{v\in V} crv_v\sum_{(d,m)\in U}(ta_{dm}-ta_{d(m-1)})+\sum_{v\in V}\sum_{c\in C_v} cc_{vc}z_{vc} \quad (1)$$

Constraints:

$$\sum_{(d,m)\in U}\sum_{(j,n)\in N} x_{jndmv}-\sum_{(d,m)\in U}\sum_{(j,n)\in N} x_{dmjnv}\,.\,u_{iv}-r_{ik}=0,\ u_{iv}=\begin{cases}1, DWT_v\le DWT_i\\0, DWT_v> DWT_i\end{cases}, r_{ik}=\begin{cases}1, s_{imk}+q_{imvkc}\ge CR_{ik}(PH-te_{im}\\0, s_{imk}+q_{imvkc}< CR_{ik}(PH-te_{im}\end{cases}, \forall v\in V,\forall k\in K,\forall(v,d,m)\in VxU,\forall(v,i,m,j,n)\in VxN \quad (2)$$

$$\sum_{(i,m)\in N}\sum_{(d,n)\in U} x_{imdnv}=\begin{cases}1, I_{imvck}=0\\0, I_{imvck}>0\end{cases}, \forall v\in V,\forall(v,d,n)\in VxU,\forall(v,i,m)\in V_h xN \quad (3)$$

$$\sum_{(j,n)\in N} x_{jnimv}-\sum_{(j,n)\in N} x_{imjnv}\,.\,u_{iv}=0\ ,\forall(v,i,m)\in VxN, i\ne j \quad (4)$$

$$\sum_{v\in V}\sum_{(j,n)\in N} x_{jnimv}+y_{im}=1,\forall(i,m)\in N, i\ne j \quad (5)$$

$$y_{im}-y_{i(m-1)}\ge 0,\forall(i,m)\in N, m\ne 1 \quad (6a)$$

$$y_{dm}-y_{d(m-1)}\ge 0,\forall(d,m)\in U, m\ne 1 \quad (6b)$$

$$x_{dmjnv}\in\{0,1\},\forall v\in V,\forall(d,m,j,n)\in A_v \quad (7a)$$

$$x_{imjnv}\in\{0,1\},\forall v\in V,\forall(i,m,j,n)\in A_v \quad (7b)$$

$$u_{iv}, r_{ik}\in\{0,1\},\forall k\in K,\forall v\in V,\forall i\in H_{Tv} \quad (7c)$$

$$y_{im}\in\{0,1\},\forall(i,m)\in N \quad (7d)$$

$$z_{vc}=\begin{cases}1, by_{dmkvc}-by_{d(m+1)k_v vc}=0\\0, by_{dmkvc}-by_{d(m+1)k_v vc}\ne 0\end{cases},\forall v\in V,\forall(d,m)\in U,\forall k_v\in K_{v,}\forall(k,c)\in K_v xC_v, k_v\ne k \quad (8)$$

$$\sum_{k\in K} by_{dmvck}\le 1,\forall v\in V,\forall(d,m)\in U,\forall(k,c)\in K_v xC_v \quad (9)$$

$$\sum_{c\in C_v} by_{dmvck}\le|C_v|.\left(1-by_{dmvc_c k_k}\right),\forall v\in V,\forall k\in K,\forall(d,m)\in U,\forall c_c\in C_v,\forall k_k\in K_v,\forall(k,c)\in K_v xC_v \quad (10)$$

$$z_{vc}\in\{0,1\},\forall v\in V,\forall c\in C_v \quad (11a)$$

$$by_{dmkvc}\in\{0,1\},\forall v\in V,\forall(d,m)\in U,\forall(k,c)\in K_v xC_v \quad (11b)$$

$$\sum_{v\in V} xy_{imkv}=1,\forall k\in K_v,\forall(i,m)\in N \quad (12)$$

$$x_{imjnv}\left(I_{imvkc}-J_{ik}q_{jnvkc}-I_{jnvkc}\right)=0,\forall v\in V,\forall(i,m,j,n)\in N,\forall(k,c)\in K_v xC_v, i\ne j \quad (13)$$

$$x_{dmjnv}\left(J_{ik}q_{dmvkc}-I_{jnvkc}\right)=0,\forall v\in V,\forall(d,m)\in U,\forall(j,n)\in N,\forall(k,c)\in K_v xC_v, j\ne d \quad (14)$$

$$\sum_{c\in C_v}\sum_{k\in K_v} q_{dmvkc}=\sum_{(i,m)\in N}\sum_{k\in K_v} xy_{imkv}QS_{ik}+\left(\sum_{c\in C_v} CMax_{vc}-\sum_{(i,m)\in N}\sum_{k\in K_v} xy_{imkv}QS_{ik}\right),\forall v\in V,\forall(d,m)\in U \quad (15a)$$

$$q_{dmvkc}\le CMax_{vc},\forall v\in V,\forall(d,m)\in U,\forall(k,c)\in K_v xC_v, i\ne d \quad (15b)$$

$$q_{imvkc}\le CMax_{vc}\sum_{(j,n)\in N} x_{jnimv}\,,\forall v\in V,\forall(i,m)\in N,\forall(k,c)\in K_v xC_v, i\ne j \quad (16a)$$

$$I_{imvkc}\le CMax_{vc}\sum_{(j,n)\in N} x_{jnimv}\,,\forall v\in V,\forall(i,m)\in N,\forall(k,c)\in K_v xC_v, i\ne j \quad (16b)$$

$$x_{jnimv} - p_{imv} = 0, \forall v \in V, \forall (i,m,j,n) \in N \quad (17)$$
$$x_{jndmv} - p_{dmv} = 0, \forall v \in V, \forall (j,n) \in N, \forall (d,m) \in U \quad (18)$$
$$xy_{imvk} \in \{0,1\}, \forall v \in V, \forall k \in K, \forall (i,m) \in N \quad (19a)$$
$$p_{imv}, p_{dmv} \in \{0,1\}, \forall v \in V, \forall (i,m) \in N, \forall (d,m) \in U \quad (19b)$$
$$I_{imvkc}, q_{imvkc}, q_{dmvkc} \geq 0, \forall v \in V, \forall (i,m) \in N, \forall (d,m) \in U, \forall (k,c) \in K_v x C_v \quad (20)$$
$$ta_{i(m+1)} - ta_{im} \geq 0, \forall (i,m) \in N \quad (21a)$$
$$ta_{d(m+1)} - ta_{dm} \geq 0, \forall (d,m) \in U \quad (21b)$$
$$ta_{im} + TP_i p_{imv} \leq \left[\frac{II_{ik} - SMin_{ik}}{CR_{ik}}\right], \forall k \in K, \forall (i,m) \in N \quad (22)$$
$$A_i \leq ta_{im} \leq B_i, \forall (i,m) \in N \quad (23)$$
$$ta_{im} + TP_i p_{imv} + \sum_{k \in K_v} \sum_{c \in C_v} TQu_{ik} q_{imvkc} - te_{im} = 0, \forall v \in V, \forall (i,m) \in N, \forall (k,c) \in K_v x C_v \quad (24a)$$
$$ta_{dm} + TP_d p_{dmv} + \sum_{k \in K_v} \sum_{c \in C_v} TQl_{dk} q_{dmvkc} + \sum_{c \in C_v} TC_{vc} z_{vc} - te_{dm} = 0, \forall v \in V, \forall (d,m) \in U \quad (24b)$$
$$x_{imjnv}\left[te_{im} + TT_{ijv} - ta_{jn}\right] \leq 0, \forall v \in V, \forall (i,m,j,n) \in A_v \quad (25a)$$
$$x_{dmjnv}\left[te_{dm} + TT_{djv} - ta_{jn}\right] \leq 0, \forall v \in V, \forall (d,m,j,n) \in A_v \quad (25b)$$
$$x_{imdnv}\left[te_{im} + TT_{idv} - ta_{dn}\right] = 0, \forall v \in V, \forall (i,m,d,n) \in A_v \quad (25c)$$
$$ta_{im}, te_{im} \geq 0, \forall (i,m) \in N \quad (26a)$$
$$ta_{dm}, te_{dm} \geq 0, \forall (d,m) \in U \quad (26b)$$
$$s_{imk} = II_{ik} - CR_{ik}(ta_{im} + TP_i p_{imv}), \forall\, v \in V, \forall (i,k) \in (H_T - d) x K_i \quad (4.26)$$
$$SMin_{ik} \leq s_{imk} \leq SMax_{ik}, \forall (i,m,k) \in (H_T - d) x K_i \quad (4.27)$$
$$SMin_{ik} \leq s_{imk} + q_{imvkc} - CR_{ik}\big(te_{im} - (ta_{im} + TP_i p_{imv})\big) \leq SMax_{ik}, \forall\, v \in V, \forall\, c \in C_v, \forall (i,m,k) \in (H_T - d) x K_i \quad (4.28)$$
$$SMin_{ik} \leq s_{imk} + q_{imvkc} - CR_{ik}(PH - te_{im}) \leq SMax_{ik}, \forall\, v \in V, \forall (i,m,k) \in N x K_i \quad (4.29)$$
$$s_{imk} \geq 0, \forall k \in K, \forall (i,m) \in (H_T - d) \quad (4.30)$$

Equation (1) is an objective function that consists of the cost of travel, port setup, ship charter and compartment washing. Constraints (2) ensure that each ship should leave the loading port if the stock level at unloading port is not able to cover the consumption of product during specified planning horizon. Equation (3) make the empty ship should be charged back to the loading port. Constraints (4) ensure that the ships coming into the port i had to leave the port to the port j. Constraints (5) and (6) relating to the ship that visit at each port. Constraints (7) show that the routing variables are binary variable. Constraints (8) ensure that the compartment washing should be done when the product that will be loaded different from the previous load. Constraints (9) and (10) related to the compatibility of the product. Constraint (11) indicates that the variable is binary variable. Constraint (12) shows that delivery can't be split. Constraints (13) to (16) describe the number of products that loaded and unloaded which both of them should not exceed the capacity of the compartment. Constraints (17) and (18) show the port setup variable. While the equation (19) and (20) related to the type of the variable. Constraints (21) explained that the starting processing time and arrival time (m+1) should occur after m. Equation (22) explain that unloading time cannot exceed the time which inventory at each port can serve their consumption. Constraints (23) associated with the starting processing time that cannot exceed the time windows. While constraints (24) and (25) show the arrival and leave time. Constraint (26) shows that the time variable is a continuous variable. Equation (27) is used to look the stock level when the ship is ready to do the process and port setup. Constraints (28) to (30) ensure that stock levels cannot exceed the minimum and maximum limit of unloading port storage when unload product, leave the port until the end of planning horizon. The Last constraint explained that stock levels are continuous variables.

3. Hybrid Cross Entropy-Genetic Algorithm

The steps that done in this algorithm, will be described as follow:

Step 1. Define inputs and outputs

Input that will be used in this algorithm are planning horizon, cost parameters, product parameters, ship parameter, port parameters, shipping parameters and algorithm parameters. While the output are total cost, ship routes, the allocation of products, total travel time, the selected sample, the number of iterations and the computing time.

Step 2. Determine initial parameter values
Initial parameters consist of samples generated in the population (N), the ratio of elite samples (ρ), smoothing coefficient (α), the mutation parameter (Pm) and termination criteria of iteration (ε).
Step 3. Generate the initial sample
The samples that generated in the first iteration are random, while in the next iteration samples will be generated by mutation step.
Step 4. Calculate the shipment quantity
Shipment quantity is calculated by the following formula:

$$\text{Shipment Quantity } (QS_{ik}) = \left(PH - \frac{II_{ik} - SMin_{ik}}{CR_{ik1}}\right) \times CR_{ik} + (ta_{im} + TP_i) \times CR_{ik} \quad (32)$$

Where

$$c = TP_d + (DWT_i/TQl_{dk}) + TT_{di} \quad (33)$$

$$ta_{im} = c \text{ if } c - \left(\left\lfloor \frac{c}{24} \times 24 \right\rfloor\right) \geq A_i \text{ and } c - \left(\left\lfloor \frac{c}{24} \times 24 \right\rfloor\right) < B_i \quad (34)$$

(35)

$$ta_{im} = \left(\left\lceil \frac{c}{24} \right\rceil \times 24\right) + A_i \text{ if } c - \left(\left\lfloor \frac{c}{24} \times 24 \right\rfloor\right) \geq A_i \text{ and } c - \left(\left\lfloor \frac{c}{24} \times 24 \right\rfloor\right) > B_i \quad (36)$$

$$ta_{im} = A_i \text{ if } c - A_i < 0$$

Step 5. Loading products in the compartment
When performs loading activity, the product compatibility and compartment washing cost should be considered.
Step 6. Update the shipment quantity
This step ensures that the ship departed in full load conditions. This updating depends on the consumption rate at each port.
Step 7. Ship routing
This step ensures that the start time should be in the range of time windows.
Step 8. Calculate the objective function
The objective function consists of the travel cost, port setup, ship charter, and compartment washing cost.
Step 9. Select elite sample
Samples that will become elite samples are ⌈ρxN⌉ individuals. But before do this step, the solution must be sort first.
Step 10. Elitism
Elitism will be done to one sample at each iteration and the sample never performs the mutation.
Step 11. Mutation parameter update
Mutation parameters are updated with the following formula:

$$A_{it} = (1 - \alpha) * u + (A_{it-1} * \alpha) \quad (37)$$

Where *u* can be calculated by the following formula:

$$u = \frac{\overline{ze}}{2 * z_{best}} \quad (38)$$

$\overline{ze}$ is the average of elite sample and z_{best} is the best objective function at each iteration. Mutation parameter value is determined as follows:

$$Pm_{it} = \frac{A_{it}}{2} \quad (39)$$

Step 12. Select the type of mutation
The mutation that will be used in this research is swap, insertion, inversion, forward and backward mutation. The selection mechanism uses equation ⌈bilangan $random$ x 5 ⌉.
Step 13. Mutation
Mutations performed on all samples except the elitism sample.

4. Experiment and Analysis

Experiments are performed with several objectives, the first goal is to get the algorithm parameters, the second goal is to see the ability of the algorithm to get solutions for different conditions and the final goal is to see the performance of the algorithm that have been developed.

In the first experiment, the results show that best parameters that should be used in the algorithm are N=1000, ρ= 0.9 and α = 0.2. N value influences the number of samples that should be evaluated at each iteration. If the amount of N is bigger, the solution that will be resulted is better. While the parameters α and ρ affect the number of iterations. More iteration will make the solution better. However, when determining the three parameters it still needs to consider the time consuming. And for the second experiment, it can be concluded that the algorithm is able to produce solutions consistently by the changing of some conditions and meet all of the constraint.

The last experiment is conducted to compare the solutions between hybrid CEGA and hybrid TS. The purpose of this experiment was to determine the performance of the algorithm CEGA compared with other algorithms that have been used to solve similar problems. Table 3 shows the difference of solution of hybrid CEGA and hybrid TS have. The data that used in this experiment is the same as the data that used in the third experiment.

Table 3 Comparing Solutions Between CEGA and HTS

Experiment	Best Solution *Hybrid* CEGA (Rp)	Best Solution *Hybrid* TS (Rp)	Experiment	Best Solution *Hybrid* CEGA (Rp)	Best Solution *Hybrid* TS (Rp)
1a	3.655.100.000	3.684.500.000	3a	1.607.900.000	1.711.300.000
1b	2.149.300.000	2.168.800.000	3b	2.021.300.000	2.310.500.000
1c	1.562.900.000	1.739.200.000	3c	3.818.900.000	4.117.500.000
2a	2.594.400.000	2.699.900.000	4a	2.288.700.000	2.384.500.000
2b	2.692.800.000	2.807.300.000	4b	2.579.900.000	2.621.300.000
2c	2.982.000.000	3.020.800.000	4c	2.664.700.000	2.834.600.000

From the twelve experiments have been done before, hybrid CEGA produces better solutions than hybrid TS. One of the reasons is CEGA population-based while HTS individual-based. So CEGA can evaluate more sample than HTS. In addition, CEGA have mutations step that consist of swap, insertion, inversion, forward and backward mutation. This mechanism makes diversification in its solution and spread the result to larger solution space. The advantage of this step is there are more solutions can be examined. In HTS, the reason that makes the solution worse is the application of taboo list which only for a combination that produces solutions that violate the constraint. While the solution that worse than the previous solution was not included in this list. So there is the possibility of repeating the same solutions and produce non optimal solutions. In computing time side, CEGA takes longer time because in one iteration a lot of samples have to be evaluated. Except that, the computation time in CEGA also depends on the convergence of solutions that obtained. If the convergence goes slowly, the number of iterations required more and more.

5. Conclusion

The conclusion from this research is multi-product inventory ship routing problem with heterogeneous fleet models that considering the weight of the ship docked in port, product compatibility, port setup and compartment washing can be conducted. Hybrid CEGA that developed in this research is able to solve the problems consistently on several conditions, when there is a change in the number of products, number of ships, number of ports to be visited and the number of ports that can be visited by ship. And when compared with the hybrid TS algorithm, hybrid CEGA is able to give a better solution for all sets of data but the computing time is longer.

6. References

[1]. Andersson, H., Duesund, J. M. dan Fagerholt, K. 2011. Ship Routing and Scheduling with Cargo Coupling and Synchronization Constraints. *Computers & Industrial Engineering,* 61**,** pp. 1107-1116.

[2]. Chopra, S. dan Meindl, P. 2007. Supply Chain Management Strategy, Planning and Operation. Prentice Hall: New Jersey.

[3]. Siswanto, N., Essam, D. dan Sarker, R. 2011. Solving the Ship Inventory Routing and Scheduling Problem with Undedicated Compartments. *Computers & Industrial Engineering,* 61**,** pp 289-299.

[4]. Hwang, S.J. 2005. *Inventory Constrained Maritime Routing and Scheduling for Multi-Commodity Liquid Bulk.* Dissertation. Georgia: Georgia Institute of Technology.

[5]. Rusdiansyah, A. dan Nurminarsih, S. 2012. Hybrid Tabu Search for Solving Multi Product Tanker Scheduling Problem (m-TSP) Considering Product Loading Compatibility Constraint. *Industrial Engineering & Management System Conference*.

[6]. Hvattum, L. M., Fagerholt, K. dan Armentano, V. A. 2009. Tank Allocation Problems in Maritime Bulk Shipping. *Computers & Industrial Engineering,* 36**,** pp. 3051-3060.

[7]. Boer, P. T. D., Kroese, D. P., Mannor, S. dan Rubinstein, R. Y. 2005. A Tutorial on the Cross-Entropy Method. *Annals of Operations Research***,** pp. 19-67.

[8]. Laguna, M., Duarte, A., dan Marti, R. 2007. Hybridizing the Cross-Entropy Method: An Application to the Max-Cut Problem. *Computers & Operations Research,* 36, pp. 487-498.

[9]. Santosa, B., Budiman, M. A. dan Wiratno, S. E. 2011. A Cross Entropy-Genetic Algorithm for m-Machines No-Wait Job-Shop Scheduling Problem. *Journal of Intelligent Learning System and Applications,* 3**,** pp. 171-180.
[10]. Arwanto, A. 2007. Penentuan Jumlah dan Jenis Kapal Distribusi Bahan Bakar Minyak (BBM) dengan Menggunakan Metode Simulasi (Studi Kasus: PT. Pertamina UPMS VIII). Tugas Akhir. Surabaya: Institut Teknologi Sepuluh Nopember.

Biography

Budi Santosa is a professor and Head of Department at Industrial Engineering Department, Institut Teknologi Sepuluh Nopember (ITS), Surabaya Indonesia. He earned B.S. in Industrial Engineering from Institut Teknologi Bandung (ITB), Indonesia, Masters and PhD in Industrial Engineering from University of Oklahoma, USA. He has published journal and conference papers. Dr Budi Santosa has done research projects in application of methaeuristics techniques in logistics, transportation and scheduling. His research interests include optimization, metaheuristics, scheduling and data mining.