Optimizing Airline Seat Allocation using Reinforcement Programming

Muhammad Imam Isfahani, Ali Ridho Barakbah, Entin Martiana
Department of Informatics and Computer Engineering
Electronic Engineering Polytechnic Institute of Surabaya,EEPIS
Surabaya,Indonesia
isfahani@it.student.pens.ac.id, ridho@pens.ac.id, entin@pens.ac.id

***Abstract*—. Low season is always be a big problem for the airlines as the occupant hugely decreased when compared to peak season. Generally, to attract more occupant in low season the airlines will sell more discount fare but the other hand they must balancing the seat allocation to avoid loss. This paper proposed new approach to optimizing the seat allocation. This approach using new algorithm which use the basic concept from reinforcement learning called reinforcement programming. First, we using the previous data to random the initial set of solution. Afterward we enforce some rule and objective function to make initial solution move toward the best solution. Finally, solution will show the revenue and a set of seat allocation for each subclass. We compare result of our idea into several optimization algorithm. The experimental result show the effectiveness of our idea to solve this seat allocation problem.**

Keywords— Seat Inventory Management; Optimization; Reinforcement Programming; Seat Allocation

I. INTRODUCTION

Seat Inventory Management is a practice of balancing the number of seats at a low price / discount and the number of seats at a high price in order to maximize the revenue of the total seat and load factors. Seat Inventory Management is one of several important elements in the ARM (Airlines Revenue Management) commonly used by the airline to carry out the management of revenue. Many methods were developed to perform seat inventory management one of them is optimization.

In the airlines, optimization is used for balancing the allocation of the airfares/seat so they know how many ticket/seat which must be sold each class to obtain major profit. Usually, airlines sell some kind of ticket which have heterogeneous price depend of the service. Generally, we know that there are three types of ticket which sold by airlines there are Executive/First/Business Class, Economy Class and Promo class but the facts are the airlines sold more than three types of ticket in each flight which derived from this classes they often called it subclasses. Subclass using NATO phonetic codes to distinguish each other such as C for Charlie, A for Alpha and etc.

Airlines not only optimize the number of tickets which they sell to gain major profit. It depend of situation which commonly known as low season and peak season. At peak season airlines can sold the subclasses which has higher price because there are much occupant. However, when the low season come airlines can't easily sell the ticket because they got less occupant. To attract more passengers in low season airlines usually sell more promo ticket which has the lowest price.

The main problem is if the airlines too much sell the promo ticket the allocation will be unbalance and there are any chance that the airlines will get loss. To prevent this, airlines must determine the seat allocation of each subclass before they sell it. The other problem is to doing this there are many constraint and variable so that make it quite complicated moreover they do it manually.

Until recently, few airlines still doing the seat allocation manually. It will make the solutions are less accurate and take more time. Even more the airlines have more than one flight in each day. Less accurate in this case will not only impact at the revenue which they will gain but also in the seat load factor of the flight. Frequently, the airlines will get less revenue or loss if the solution is less accurate.

The EMSR method of Belobaba[2] is an heuristic which frequently used by airlines nowadays. EMSR (Expected Marginal Value) generate nested protection level in nested seat allocation model. Renwick E Curry [14] proposed a new method in seat allocation by combining the marginal seat revenue with mathematical programming. By combining these method they can handle larger problem in seat allocation. Optimal the seat allocation between two flight which has different departure time was proposed by Chen, et al [15]. It uncover the structure of optimal booking policies through four monotone switching curves. Dan Zhang, William L. Cooper [16] consider the simultaneous seat- inventory control of a set of parallel flight using the dynamic customer choice. They create a simulation-based technique to solving the stochastic optimization problem and also describe heuristics based upon an extension of a well-known linear programming formulation. Feng Y, and Xiao B [17] consider an airline seat-inventory control problem with multiple origins, one hub and one destination. This study presents a stochastic control model and develops optimal control rules. The basic model is subsequently distended to consider multiple fares on each route, time-dependent demands, and booking control on an extended network. Wen Zhao and Yu-Seng Zheng [18] claim

978-1-4673-9345-4/15/$31.00 ©2015 IEEE

that the demand for a fare class is affected not only by the current availability of lower fares but also by the possibility of future availability of them. They propose two-class dynamic seat allocation model, which has two distinctive features. The model assumes first that the discount fare cannot be reopened once closed and, second, that a fraction of the customers are flexible. Youyi Feng et al [19] study about optimal the airline booking with considering the cancelation and no-shows. They re-allocate the unboked seat with overbooking model and proposed a new optimal seat control policy. Dira Mariana [20] study about overbooking seat allocation for two parallel flights owned by the same airline with single leg route. She have combined the two parallel flights seat allocation control model and revenue management with overbooking, cancellation, and no-show model. To solve this problem, we have developed a dynamic programming model with discrete time setting. Charles AN, Nadya Sartika[10] proposed a method to determine the subclasses of an flight using BEP analysis. With this analysis they can found the flight in the same route which have highest revenue. S. L. Brumelle, J. I. McGill, [21] addresses the problem of determining optimal booking policies for multiple fare classes that share the same seating pool on one leg of an airline flight when seats are booked in a nested fashion and when lower fare classes book before higher ones. They show that a fixed-limit booking policy that maximizes expected revenue can be characterized by a simple set of conditions on the sub differential of the expected revenue function.

Our proposed idea is make a seat allocation optimization system which using reinforcement programming in order to ease the airlines to balancing the seat allocation. The variables and rules which applied on this system using airline's flight data and previous sales data, so the airline can easily adjust the optimization rule and variables depend on the requisite.

II. Basic Theory of Reinforcement Programming

Reinforcement programming algorithm is a new algorithm which using basic concept of reinforcement learning. In its implement reinforcement programming has the same behavior of reinforcement learning. Reinforcement programming also using exploration or exploitation (knowledge) rate to find solution. Each process solution will be given punishment or reward depend on the resulting solution whether is god or bad. The objective of this algorithm are get as much reward as possible to determine the best solution [1].

Fig.1 is described the system architecture of reinforcement programming algorithms:

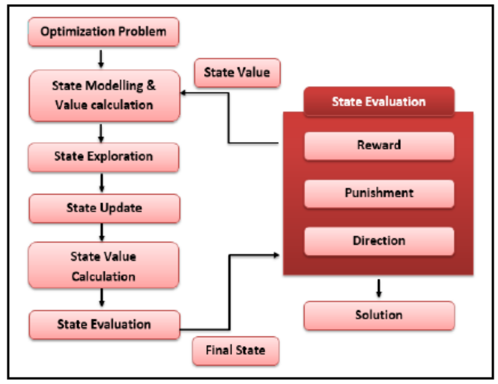

Fig. 1. RP system architecture

First, the RP will process the case by random a set of initial state and calculation of fitness value / state value calculation. Then the agent will enter the phase state exploration by random the exploration with a number between 0-1 and the action taken is determined by the exploration rate. If the number is greater than the result of random exploration rate then the agent will choose a random state. But if the number is smaller than the result of random exploration rate then the agent would choose a state based on the value of the greatest rewards. After perform action the agent will update the value of the current state and calculate the value of the new state. The value of the previous state will be used to determine reward or punishment and determines direction of further exploration. If the agent get punishment then the agent will change the direction for the next exploration step and variable reward will be updated with the formula:

$$r_p=r_p-\beta*(1-r_p). \tag{1}$$

Meanwhile, if the agent receives reward the variable reward will be updated with the formula:

$$r_p=r_p+\beta*(1-r_p). \tag{2}$$

Which r_p is state position that will be rewarded or punished and β is a variable to give impact to step that agent will take

III. DETAIL DISCUSSION

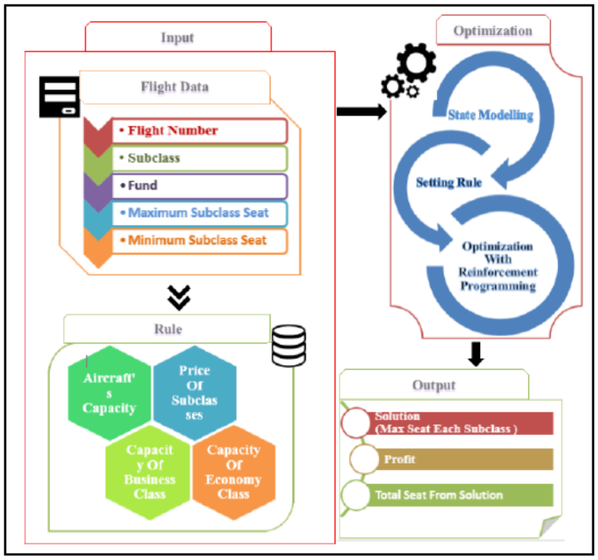

Fig. 2. System Architecture

Fig. 2 is a general overview of the system architecture which will describe the system of the application. To process cases with reinforcement programming algorithm takes some input variable used as a constraint or rule, and data flight that will be optimized.

Input variables are variables obtained from user input. A few variable can be filled with previous data according to the conditions. This variable also called flight data which airlines have. These variables are:

- Flight Number: flight number that indicates the destination of the plane.
- Subclass: subclasses which specified in the flight. In this case the maximum subclass in one flight is 9 with details 1 promo class, 6 economic class and 2 business class.
- Fund: total cost incurred in flight.
- Maximum Subclass Seat: maximum number of seats each subclass. Can use the previous sales data.
- Minimum Subclass Seat: minimum number of seats each subclass. Can use the previous sales data.

Rule variable is variable obtained from the database with the parameters of the variable input. This variable will act as a constraint in this optimization process. These variables are:

- Aircraft Capacity: the maximum aircraft's capacity. This variable is obtained through the Flight Number parameter from data flight. Each Flight Number has different type and capacities.
- The price of each subclass: the price of each subclass which input to the system. This variable is the main variable in optimization because this price is one of the major constraints in the fitness function to determine the number of seats each subclass.
- Economy Class Capacity: This variable is a constraint that limits the maximum number of seats for economy and promo class (Q, T, V, M, K, B, Y).\
- Business Class Capacity: This variable is a constraint that limits the maximum number of seats for Business Class (J, C).

Below is described in detail the implementation of the algorithm in this case:

1. Optimization Problem: This phase is the phase where we define the problems and define and constrain state of optimization which we will calculate. In this phase we model the subclass into the state. The Fig.3 below is an example of modeling subclasses which sell by airlines :

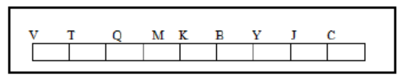

Fig. 3. State Modelling

In the model above every state has nine genes that is a representation of each subclass. After we modeling the state next we need to define the constraint for the problem. Here below the constraint:

   - The total value of the state is not more than the maximum capacity of the aircraft.
   - Total value of executive class (J / C) should not be more than 12
   - Total value of economy class and promos (Q, V, T, M, Y, K, B) not more than 84/150 depending on the type of aircraft.
   - Results of the optimization of the total price must be greater than capital or generating profits
   - The value of each stat should not be more than the maximum value or less than a predefined minimum value.

2. State Modeling & Value calculation: after we modeling the state we determine the objective function of the optimization process in this case the fitness function is :

$$f(x) = (Seat*Price)+(0.1*PromoSeat) \quad (3)$$

in this phase we also initiate few variable such as reward, step and direction which is the main variable in reinforcement programming.

3. State Exploration: in this phase we random a number between 0-1. if the results exceed the random exploration rate it will be exploratory, which means the state will be chosen at random and if the result is less random than exploration rate it will be done in the exploitation of the state where the election is determined by the greatest reward.

4. State Update: This phase we update the value of the state and stored in a new state.
5. State value calculation: this phase we calculated the new fitness value of the new state
6. State Evaluation: This phase compares the fitness values of the old state and the new state. If the new state has a higher fitness values will be given a reward and if the smaller will be given a punishment and if equal the direction of state will be updated.
7. This process will repeated until a condition fulfilled.

Below will explain the output from the optimization process:

- Total Seat each Subclass: This is the main solution of search optimization. The main goal is to find the right allocation in each subclass, so they not only get the profit but also multiply sales the promo subclass.
- Profit: revenue which obtained when using the allocation according to the set of solution
- Total seat optimization results: total seat of optimization results. It could be a total seat lower than the aircraft's capacity has met the conditions of profit so that the rest of the seat can be freely distributed into any subclass.

IV. Experiment and Analysis

The experiment has been tried using real data flight in several optimization algorithm such as genetic algorithm (GA), particle swarm optimization (PSO), simulated annealing (SA) and ant colony optimization (ACO). This experiment not only show the fitness value but also the profit which generated from the solution. This experiment using operational cost which used in low cost courier airlines and the average ticket price Rp. 700.000,00.

The experiment using 5, 10, 15 seconds iteration. In each algorithm runs in 10 times of experiment. The accuracy of current state will determine by the biggest fitness value. Higher the fitness value which yielded by solutions indicates better seat allocation and revenue.

GA algorithm used 10 individual in one population and the operator of cross over are direct switch and mutations are 10% from maximum seat allowed. Fig. 4 below will show average performance of genetic algorithm in 5, 10, 15 seconds iteration. Fig. 5 show the average profit which will obtained when using the solution from this algorithm. We also perform the experiment profit in 5, 10 15 seconds iteration.

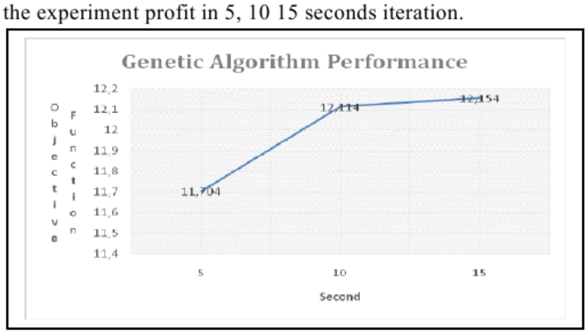

Fig. 4. GA Average Performance

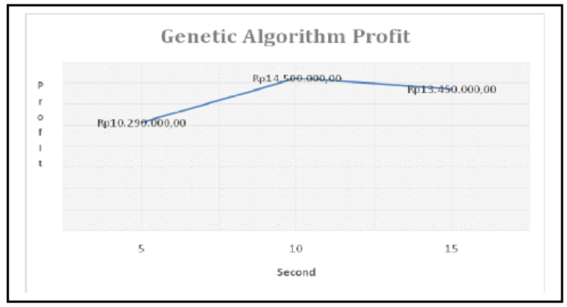

Fig. 5. GA Average Profit

PSO algorithm used 10 particles in one swarm and limit the maximum velocity is 10% from maximum seat allowed. Value of learning factor of this experiment are c1=2 and c2=2. Fig. 6 below will show average performance of particle swarm optimization in 5, 10, 15 seconds iteration. Fig. 7 show the average profit which will obtained when using the solution from this algorithm. We also perform the experiment profit in 5, 10 15 seconds iteration.

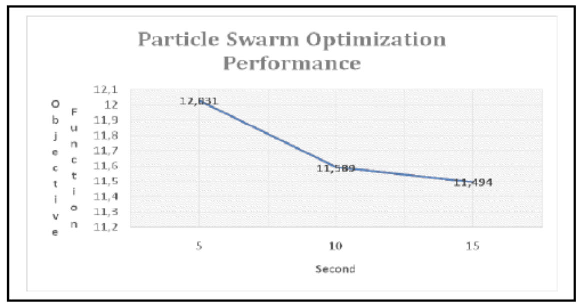

Fig. 6. PSO Average performance

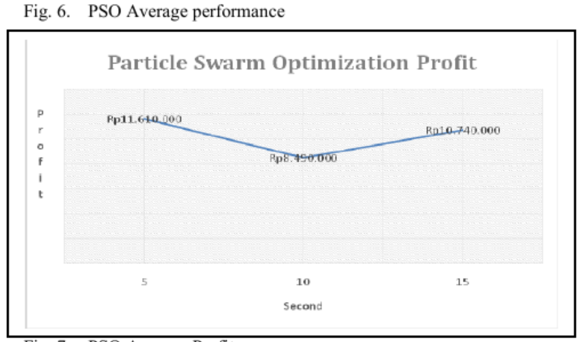

Fig. 7. PSO Average Profit

SA algorithm used random to select the state and the new state are produced by add selected state with 10% from maximum seat allowed. Fig. 6 below will show average performance of simulated annealing in 5, 10, 15 seconds iteration. Fig. 7 show the average profit which will obtained when using the solution from this algorithm. We also perform the experiment profit in 5, 10 15 seconds iteration.

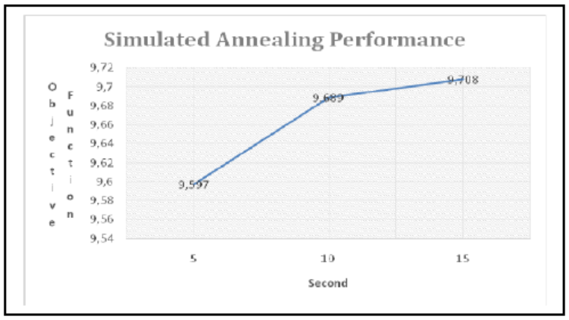

Fig. 8. SA Average Performance

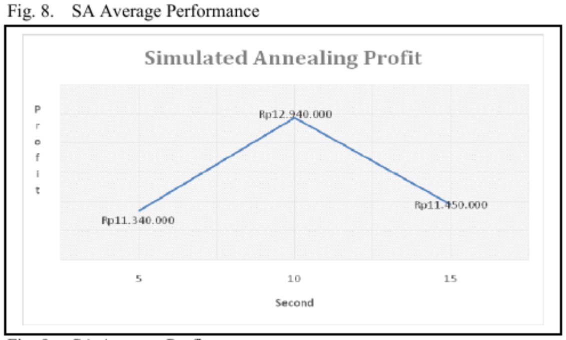

Fig. 9. SA Average Profit

ACO algorithm used probabilistic to determine the edge selection whether the best pheromone or roulette function. All ants will perform local pheromone update to decrease pheromone and only the best ant will perform global pheromone update to increase pheromone. Fig. 6 below will show average performance of ant colony optimization in 5, 10, 15 seconds iteration. Fig. 7 show the average profit which will obtained when using the solution from this algorithm. We also perform the experiment profit in 5, 10 15 seconds iteration.

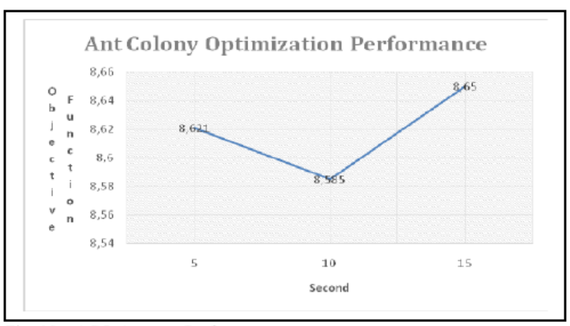

Fig. 10. ACO Average Performance

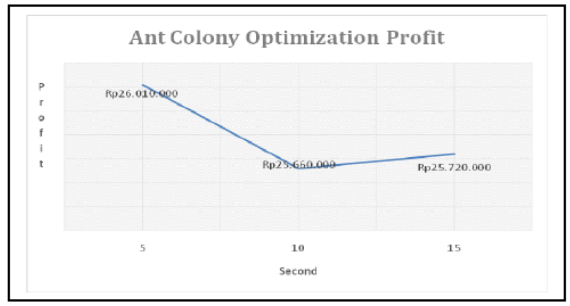

Fig. 11. ACO Average Profit

As for RP the value of variable that used in experiments are: miu =0.005 , beta = 0.5 , exploration rate = 0.5. Fig. 6 below will show average performance of reinforcement programming in 5, 10, 15 seconds iteration. Fig. 7 show the average profit which will obtained when using the solution from this algorithm. We also perform the experiment profit in 5, 10 15 seconds iteration.

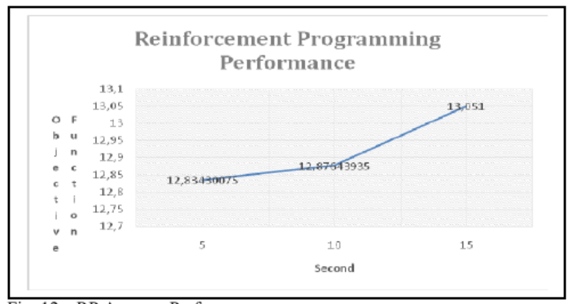

Fig. 12. RP Average Performance

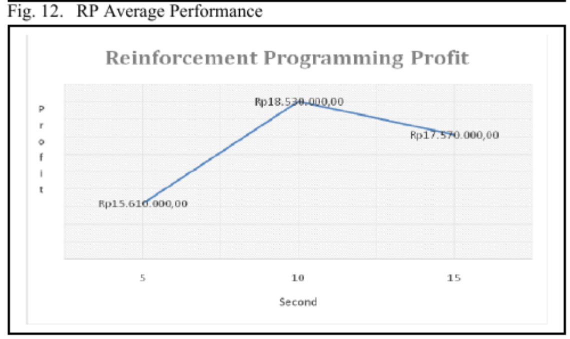

Fig. 13. RP Average Profit

Fig.14 will shown the graph which compare all algorithm performance and Fig.15 shown the graph which compare all algorithm profit.

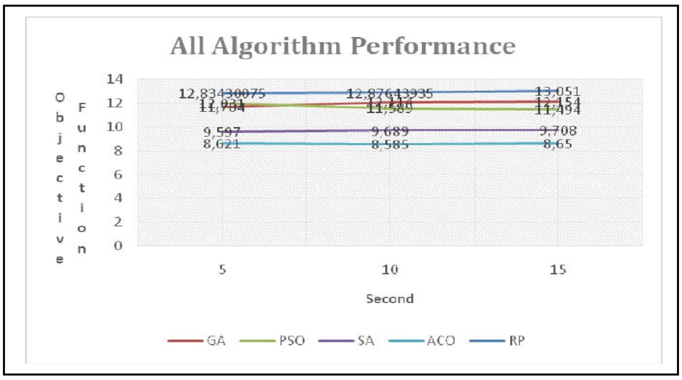

Fig. 14. All Algorithm Performance

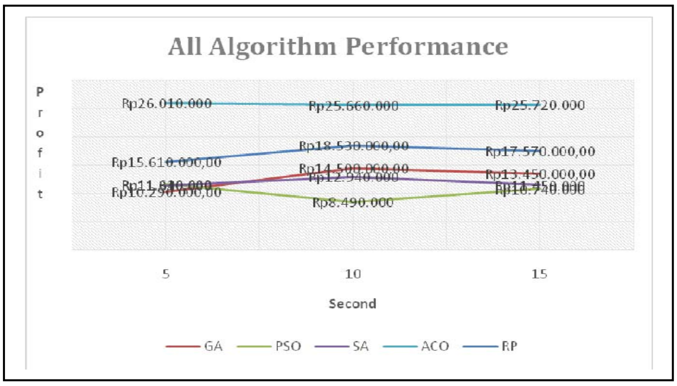

Fig. 15. All Algorithm Profit

From Fig. 15 above we know that the reinforcement of programming had better average fitness value compared than four other algorithms. Reinforcement programming algorithm is the most frequently reach the global optimum compared to four other algorithms that are often trapped in local optimum.

Fig. 12 shown the comparison of average profit all algorithm. As we can see ACO has the best profit compared to the other algorithm. Although ACO have the best profit but ACO has the worst fitness value compared to the others. In this paper the best algorithm is determined by fitness value and the profit. From this experiment we know that reinforcement programming has the best average fitness value and the second average in profit.

V. CONCLUSION

Nowadays, many airlines using subclasses as their pricing concept. With using subclasses the airlines can sell ticket with various prices. In other hand it will make balancing seat allocation would be a problem even less when low season come. We can use optimization to estimate the allocation of seat for each subclass before selling the ticket.

Optimization will greatly impact to the seat allocation and revenue of the airlines while low season come. With optimization, airlines can sell promo ticket without fear of losing. This paper offers a solution using reinforcement programming algorithm which also compared with several optimization algorithm. The experimental evaluation scheme was used to provide a common base of performance and profit. Finally, when comparing the experimental result by using reinforcement programming, genetic algorithm, particle swarm optimization, simulated annealing and ant colony optimization it observed clearly that reinforcement programming has the best performance among algorithm and second best profit. So among the others algorithm it clearly seen that reinforcement programming has the best performance of all. This algorithm suitable for implemented in this case. In the future we will extend our system with forecasting system which processed during the optimization.

REFERENCES

[1] Ali Ridho Barakbah , Irene Erlyn Wina Rachmawan, Ira Prasetyaningrum ,Yuliana Setiowati , Reinforcement Programming: A Function Based Reinforcement Learning, The Third Indonesian-Japanese Conference on Knowledge Creation and Intelligent Computing (KCIC) 2014, March 25-26, 2014, Malang, Indonesia.

[2] Peter Paul Belobaba , Air Travel Demand and Airline Seat Inventory Management , MIT ,1987.

[3] Soekartawi, Linear Programming Teori dan Aplikasinya Khususnyadalam Bidang Pertanian. Jakarta: PT. Raja Grafindo Persada., 1995.

[4] Mulyono, S., Operation Research. Jakarta: Lembaga Penerbit FakultasEkonomi Universitas Indonesia,Penanggunalan Bencana, 1991

[5] Koopman Nicholson, W., Teori Ekonomi Mikro: Prinsip Dasar dan Pengembangan.Cetakan Ketiga. Terjemahan Deliarnov, Jakarta: Raja Grafindo Persada, 1994.

[6] Supranto, J., *Riset Operasi: Untuk Pengambilan Keputusan*, Jakarta:Lembaga Penerbit Fakultas Ekonomi Universitas Indonesia, 1988

[7] Darsono, R, Tarif Dokumen Pasasi ,Jakarta : Alfabeta, 2004

[8] ticketinginfo.blogspot.com
(Last Updated – 21 Januari 2015)

[9] Charles AN, Nadya Sartika, Penentuan Subclasses Berdasarkan Tipe Pesawat, 2014

[10] http://www.swarmintelligence.org
(Last Updated – 24 Juni 2015).

[11] Miftahul Huda, Achmad Basuki, dan Tri Budi Santoso, Penyelesaian alokasi kanal radio dinamis menggunakan metoda simulated annealing(SA), 2003

[12] http://www.scholarpedia.org/article/Ant_colony_optimization
(Last Updated – 24 Juni 2015).

[13] https://en.wikipedia.org/wiki/Ant_colony_optimization_algorithms
(Last Updated – 24 Juni 2015).

[14] Renwick E Curry , Optimal Airline Seat Allocation With Fare Classes Nested by Origins and Destination, 2001

[15] Chen, S., Gallego, G., Li, M. Z. F., and Lin, B. , Optimal Seat Allocation for Two Flight Problems with a Flexible Demand Segment, European Journal of Operation Research, 201(3), 2010, pp. 897-908.

[16] Zhang, D, and Cooper, W. L., Revenue Management for Parallel Flights with Customer Choice Behavior. Operations Research, 53(3), 2005, pp. 415-431.

[17] Feng, Y, and Xiao, B., A Dynamic Airline Seat Inventory Control Model and Its Optimal Policy. Operation Research, 49(6), 2001, pp. 938-948.

[18] Wen Zhao, Yu-Sheng Zheng, A Dynamic Model for Airline Seat Allocation with Passenger Diversion and No-Shows,2001

[19] Youyi Feng, Ping Lin, Baichun Xiao,Optimal Airline Booking Control with Cancellations,Stochastic Modeling and Optimization 2003, pp 395-427

[20] Dina Mariana,Dynamin Programming Model to Determine Optimum Seat Allocation for Two Pararel Flights with Cancelation and No Show ,2010

[21] S. L. Brumelle, J. I. McGill, Airline Seat Allocation with Multiple Nested Fare Classes, 1993.