

Development of a Genetic Algorithm for Advertising Time Allocation Problems

Reza Alaei¹, Farhad Ghassemi-Tari^{2*}

¹ PO.Box: 53816-14497, Tehran, Iran
reza.alaei@yahoo.com

² Department of Industrial Engineering, Sharif University of Technology, Tehran, Iran
ghasemi@sharif.edu

ABSTRACT

Commercial advertising is the main source of income for TV channels and allocation of advertising time slots for maximizing broadcasting revenues is the major problem faced by TV channel planners. In this paper, the problem of scheduling advertisements on prime-time of a TV channel is considered. The problem is formulated as a multi-unit combinatorial auction based mathematical model. This is an efficient mechanism for allocating the advertising time to advertisers in which the revenue of TV channel is maximized. However, still this problem is categorized as a NP-Complete problem. Therefore, a steady-state genetic algorithm is developed for finding a good or probably near-optimal solution, and is evaluated through a set of test problems for its robustness. Computational results reveal that the proposed algorithm is capable of obtaining high-quality solutions for the randomly generated real-sized test problems.

Keywords: Combinatorial Optimization, TV Advertising Allocation Problem, Combinatorial Auctions, Genetic Algorithms.

1. INTRODUCTION

Commercial advertising on television is the main source of revenue for TV channels. It is a billion-dollar industry which brings in a large amount of the revenue for advertisement-based broadcasters. One of the major problems faced by TV channel planners is to allocate the advertising time to advertisers in order to maximize revenue. The problem is difficult due to the limitation of time slots for advertising. Commercial systems currently used by television companies to schedule advertisements are limited in the quality of the end result and flexibility to the broadcaster. In fact, different broadcasters view the problem in different ways. Some advertising requesters require the commercial to be seen by a given number of people, and the broadcaster will schedule the commercial to achieve this goal. Others want to have the broadcast seen by a certain demographic group or certain socio-economic classes of people. Some customers may want their commercials to be at the start/end of a given advertising break and others may want the commercial to be associated with a given TV program.

* Corresponding Author

In other hand television companies view the problem in terms of their incomes and their benefits. Therefore, we limit the problem goals on the television view and formulate the problem to gain the advantages of a profitable and efficient resource allocation mechanism called multi-unit combinatorial auction. Also we restricted the scheduling time to prime-time because of its greater utility for advertisers and consequently, its greater revenues for TV companies.

Optimization techniques have been successfully employed in solving various decision problems. However, only a few studies exist that addresses the optimization of the advertising allocation problem on television industry. The research studies dealing specifically with TV breaks advertising allocation problem is sparse. Brown (1969) described various hardships in allocating the advertising time to advertisers manually, and he developed an algorithm for exchange of advertisements of different lengths between advertising breaks to gain space for other advertisements.

Mihiotis and Tsakiris (2004) studied the problem of finding the best possible combination of placements of an advertisement (which channel, when, and how often) with the objective of the highest rating and subject to the limitation of the advertising budget. The problem was modeled as an integer program and a heuristic algorithm was designed for finding a good solution.

Bollapragada et al. (2002 & 2004) considered the commercial scheduling problem of a single advertiser. The problem was formulated as an integer program and solved sequentially for each advertiser. Bollapragada, Bussieck and Malilk (2004) then studied the problem of scheduling commercials over a specific period so that the airing of the same commercials are spread as evenly as possible and formulated this problem in the form of the integer programming models. They employed a branch-and-bound solution approach for obtaining the solution and also developed a heuristic approach for the multiple-air campaign problem.

Jones (2000) presented the advertising allocation problem as an example of designing incompletely specified combinatorial auctions in which hundreds of advertisers can submit combinatorial bids for allocating their commercials in the advertising slots. The problem was formulated as an integer programming model, and employed heuristics based on constraint programming to find a set of feasible solutions for their mathematical model. Based on his work, Zhang (2006) studied the problem of selling the advertising time to advertisers. He proposed a two step hierarchical method to find solutions for the problem. His approach starts with selecting advertisers and assigning them to TV programs and ends with allocating the advertising time to the selected advertisers in a program. The problem corresponding to the first step was solved using column generation method.

Benoist et al. (2007) studied a particular challenge faced by French satellite television. In their work, advertisements are sold as packages, rather than as individual spots which is called the TV-break packing problem. Various solution methodologies have been proposed for this problem. Brusco (2008) also used the branch and bound method for solving the smaller scale of this problem and the simulated annealing heuristic approach for the real size problems. The developed heuristic approach has shown a significant improvement in terms of computation time and solution quality.

Kimms and Muller-Bungart (2007) described a planning problem at a broadcasting company where advertisers place orders for advertisements and their airdates are not fixed by the advertisers. The TV channel has to decide simultaneously which orders to accept or to reject and when advertisements from accepted orders should be broadcasted. They formulated the problem as a mathematical model and presented several heuristics to find solutions for the problem.

Pereira et al. (2007) developed a decision support system to plan the best assignment for the weekly promotion space of a major Portuguese TV station. The aim of this heuristic-based scheduling software system was to maximize the total viewing for each product within its target audience while fulfilling a set of constraints defined by advertisers.

Wuang et al. (2010) presented an ant colony optimization (ACO) heuristic for establishing a mechanism for solving the problem of scheduling television advertisements by considering customer requirements, relevant laws and regulations, and the need to fill all available advertising time. In the proposed approach, the scheduling mechanism and ACO heuristics are developed separately, allowing user to vary parameters of ACO heuristic and flexibly adjust the scheduling criterion. Also, Mao et al. (2011) proposed an ACO algorithm to optimize the sum of products of revenue and advertisers' credit information in TV advertising and evaluated it using real data in the Japanese TV advertising market.

In this paper, the TV advertising allocation problem is formulated based on combinatorial auction mechanism in which the TV channel sells the time available in advertising breaks on prime-time to advertisers in order to maximize revenue.

The proposed binary integer program is a form of winner determination problem in multi-unit combinatorial auctions (Cramton, 2005). Review of the literature reveals that the proposed model is categorized in the class of NP-Complete combinatorial optimization problems and we developed a genetic algorithm to efficiently find good or probably near-optimal solutions for the problem.

2. COMBINATORIAL AUCTION MECHANISM

Combinatorial auctions are those auctions in which multiple varieties of items are concurrently sold and bidders can place bids on combinations of items rather than just on single items. Such combinatorial bidding is more desired whenever bidding items are complements or substitutes of each other, at least for some of the bidders. Items are called complements items, when the utility of a set of items is greater than the sum of utilities for the individual items. For example, a pair of gloves is worth more than the value of a left glove alone plus the value of a right glove alone. In such cases, the combinatorial bidding allows the bidders to more clearly express their complex preferences, and more often allowing the auction to achieve higher revenue.

Combinatorial auctions have been used in many allocation problems, such as truckload transportation (Ledyard et al., 2002), airport slot allocation (Rassenti et al., 1982), industrial procurement (Bichler et al., 2006), and spectrum auctions (Cramton, 2005). Additionally, combinatorial auctions serve as a common abstraction for many resource allocation problems in decentralized computerized systems such as the Internet, and may serve as a central building block of future electronic commerce systems. We took the advantages of this type of auction and implemented it to the problem of allocating advertising time on television channels.

A combinatorial auction is described by a set of rules. These rules define the winner determination procedure and the amount of money that winners have to pay. Also, they can restrict the feasible bids. The TV channel as auctioneer sells its available time in advertising breaks on prime-time to advertisers by considering several rules that are listed below:

- The length of advertisements and advertising breaks is measured by a time unit that is composed of particular number of seconds (here, 15 seconds).

- A combinatorial bid includes the advertiser's required time units of different advertising breaks together with amount of money that he is interested to pay.
- Each of the time units available in advertising breaks has a minimum value for TV channel that is unknown for advertisers. Therefore, the amount of money that an advertiser has to pay for a combination of time units should exceed its minimum value, otherwise the TV channel rejects his combinatorial bid and requests him to adjust it.
- With the aim of giving more opportunity for advertisers to express their preferences, each advertiser can place more than one but limited number of bids on time units of advertising breaks on prime-time such that at most one bid of each advertiser will be accepted.

3. MATHEMATICAL MODEL

Suppose that a TV channel decides to have r advertising breaks in timetable of programs on prime-time, and q_k is the number of time units available in break k . Each advertiser can submit exactly n combinatorial bids on time units of advertising breaks, and there are m advertising requests.

The bid j of advertiser i is represented by (q_{ij}, p_{ij}) in which, $q_{ij} = (q_{ij1}, q_{ij2}, \dots, q_{ijr})$ is the vector of his requested time units from advertising breaks, and p_{ij} is the amount of money that he is interested to pay for its acceptance.

By defining the binary decision variables,

$$x_{ij} = \begin{cases} 1, & \text{if bid } j \text{ of advertiser } i \text{ is accepted} \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

the TV advertising allocation problem with the objective of maximizing the revenue of TV channel is formulated in the form of a binary integer program as (2) in which the first set of constraints ensure that the sum of requested time units from an advertising break in accepted bids do not exceed the number of time units available in that break and the second constraints guarantee that at most one bid of each advertiser is accepted.

$$\begin{aligned} \max Z &= \sum_{i=1}^m \sum_{j=1}^n p_{ij} x_{ij} \\ \text{s.t.} \\ \sum_{i=1}^m \sum_{j=1}^n q_{ijk} x_{ij} &\leq q_k, \quad k = 1, \dots, r \\ \sum_{j=1}^n x_{ij} &\leq 1, \quad i = 1, \dots, m \\ x_{ij} &\in \{0, 1\}, \quad i = 1, \dots, m ; j = 1, \dots, n \end{aligned} \quad (2)$$

The model is a form of winner determination problem in multi-unit combinatorial auctions in which there are exclusive OR constraints (second constraints) between some of bids. This kind of problem is categorized in the class of NP-Complete combinatorial optimization problems as proved in (Sandholm et al., 2002). We employed a genetic algorithm to efficiently find a good or probably near-optimal solution for the problem.

4. GENETIC ALGORITHM FOR THE PROBLEM

A genetic algorithm (GA) is an intelligent probabilistic search algorithm which can be applied to a variety of combinatorial optimization problems. The theoretical foundations of GAs were originally developed by Holland (1975). The basic steps of a simple GA are shown below:

1. Generate an initial population.
2. Evaluate fitness of individuals in the population.
3. Select parents from the population.
4. Recombine parents to produce children.
5. Evaluate fitness of the children.
6. Replace some or all of the population by the children.
7. If the solution is satisfactory, stop, otherwise go to step 2.

We developed a genetic algorithm by modifying the above mentioned simple GA, to incorporate specific knowledge of the problem's structure. The proposed GA for the problem of TV advertising allocation can be described as follows.

Representation and Fitness Function

A solution for the problem is represented by a chromosome of length m such that each gene can take a value in $\{0,1,2,\dots,n\}$. Since each gene refers to an advertiser, its value refers to the advertiser's accepted bid, if the gene has a nonzero value. If the gene has a zero value, then none of the advertiser's bids has been accepted.

This coding is used to automatically satisfy the exclusive OR constraints. If a chromosome represents an infeasible solution, it is converted to a feasible one by a repair operator. Consequently, the fitness value of each individual equals to its objective function value.

Initialization

For constructing each of 25 non-duplicate initial random feasible solutions, a constructive algorithm with computational complexity of $O(mnr)$ is applied that repeatedly randomly select an advertiser in each iteration and accepts one of his bids at random if the constructed solution until previous iteration remains feasible. This algorithm terminates when the remained advertisers cannot be added to the set of winners.

Parent Selection and Reproduction

Parent selection is the task of assigning reproductive opportunities to each individual in the population. We use binary tournament method to select two parents who will have a child.

The selected parents are experienced the reproduction phase that includes the crossover, mutation and repair operators. The uniform crossover with rate 0.8 and a particular mutation operator with rate $1/m$ are adopted for this phase.

After crossover operator, the mutation operator replaces the value of a randomly selected gene in the child solution with a value in $\{0,1,2,\dots,n\}$ that has minimum frequency in the selected gene. This is done in order to preserve the population diversity during the search.

In order to ensure feasibility of the child solution, a repair operator based on a greedy algorithm was applied. The following criterion is considered to accept the most promising bids:

$$\pi_{ij} = p_{ij} / \sum_{k=1}^r q_{ijk} \quad (3)$$

The repair operator with computational complexity of $O(mnr)$ examines each variable in increasing order of π_{ij} and if the value of gene i is to be j , changes it to zero until feasibility is achieved. This is done in such a way that the effect of mutation operator is not eliminated.

Replacement and Termination

After reproduction phase, if the produced feasible solution is non-duplicate and its fitness is greater than the average fitness of population, it is replaced by an individual with the lowest fitness value. The GA is terminated when a particular number of non-duplicate child solutions have been generated.

The major steps of the developed GA for the advertising allocation problem is illustrated in Figure 1.

5. COMPUTATIONAL RESULTS

We designed a computational experiment for evaluating the computational time and the solution quality of the developed genetic algorithm. To conduct this computational study, we generated a set of random test problems with different sizes.

We classified our test problems to 2 different categories. The first category (one day schedule from 19:00 to 24:00) contains a set of test problems in which there are 25 advertising breaks ($r = 25$) and the second category (two days schedule) contains 50 advertising breaks ($r = 50$). In each category we generated 8 scenarios of test problems with different sizes. In these test problems, the number of advertisers (m) is 100, 150, 200, and 250. Also, the number of combinatorial bids (n) for each advertiser is 5, and 10.

We then employed the following random generation mechanism to generate 10 test problems for each scenario:

- **Step 1.** For each advertising break, generate a random number of time units as its length from $\{12,16,20\}$ that corresponds to $\{3,4,5\}$ minutes.
- **Step 2.** For each advertiser, generate a random number of time units from $\{1,2,3,4\}$ as his advertisement length (l);
- **Step 3.** Considering the number of advertising breaks (r), and the advertiser's ad length (l), generate a random number as the frequency (f) of advertisement in each of his n combinatorial bids as follows:

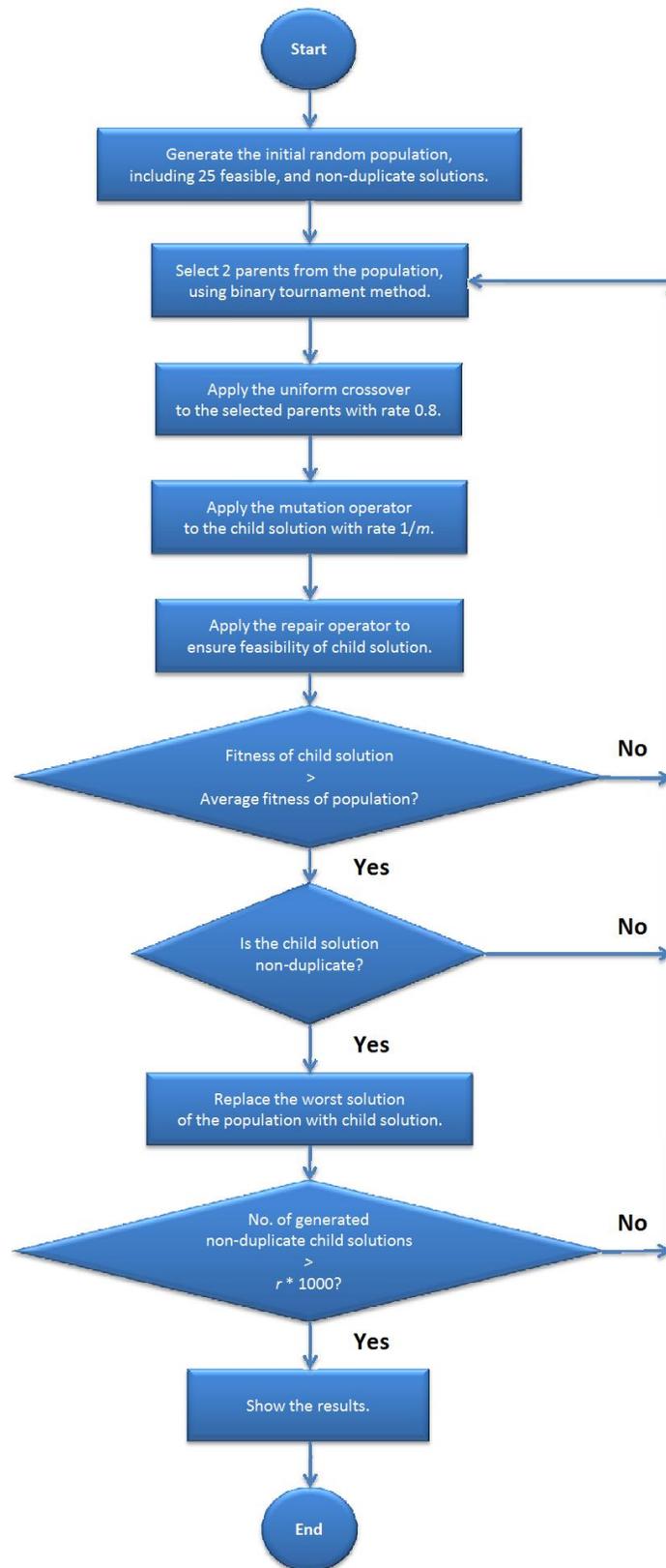


Figure 1 The flowchart of GA heuristic

$$f = \frac{r}{25} \times \text{Random number from } \begin{cases} \{4,5,6\} & l=1 \\ \{3,4,5\} & l=2 \text{ or } 3 \\ \{2,3,4\} & l=4 \end{cases} \quad (4)$$

- **Step 4.** Generate each of n combinations (q_{ij}) of an advertiser as a random permutation of l (f times), and 0 ($r-f$ times).
- **Step 5.** Considering the utility vector (U) for each of 25 advertising breaks,

$$U^t = [50 \ 50 \ 60 \ 60 \ 70 \ 70 \ 80 \ 80 \ 90 \ 90 \ 100 \ 100 \ 100 \ 100 \ 100 \ 90 \ 90 \ 80 \ 80 \ 70 \ 70 \ 60 \ 60 \ 50 \ 50] \quad (5)$$

and generate the advertiser's suggested price (p_{ij}) for each of his combinations as a function of the inner product of q_{ij} , and U as follows:

$$p_{ij} = \begin{cases} \lfloor q_{ij} \cdot U \times (1 + rand) \rfloor & r = 25 \\ \lfloor q_{ij} \cdot [U, U] \times (1 + rand) \rfloor & r = 50 \end{cases} ; \quad i = 1, 2, \dots, m, \quad j = 1, 2, \dots, n \quad (6)$$

In this equation, the symbol " $\lfloor \cdot \rfloor$ " is the function returns the first integer number that is less than or equal to the expression, and $rand$ is a uniform random number in (0,1).

To evaluate the solution quality of the developed genetic algorithm, we coded the GA in MATLAB and solved 10 test problems of each scenario. The GA was run 3 times for each test problem and each run terminated when $r \times 10^3$ non-duplicate child solutions had been generated. We also solved the LP relaxation of the same test problems.

Since the optimal solution of the binary integer program (BIP) of these sizes can not be obtained, we used the solution of the LP relaxation of BIP. It is clear that the optimal solution of the LP relaxation is an upper bound of BIP. Therefore for measuring the degree of closeness of the GA solutions to the optimal solution, we measured the deviation of the GA solution from an upper bound of the optimal solution. To conduct this measuring process, we calculated an index that we called it " $Ratio$ ". The $Ratio$ is calculated through dividing the GA's solution value (Z_{GA}) by the optimal value of the LP-relaxation problem (Z_{LP}). Mathematically the $Ratio$ is defined by:

$$Ratio = \frac{Z_{GA}}{Z_{LP}} \quad (7)$$

Table 1 and Table 2, demonstrate the computational results of the 8 test problem scenarios with $r = 25$, and $r = 50$, respectively. In these tables, we incorporated the best $Ratio$ among the 3 runs of the test problems. From these tables, it can be seen that, for the worst case, the GA solutions of the test problems are deviated by less than 10% of the upper bound of the optimal solutions. This concludes that the developed genetic algorithm can obtain the optimal and/or near optimal solutions of the considered problem.

The computational results also show that by increasing the number of combinatorial bids for each advertiser, the value of average *Ratio* does not decrease. This fact is even more encouraging since it shows that the quality of the GA solution may not be deteriorated as the problem size is increased.

The fourth column of these tables shows the run time of obtaining the best solution of the GA. The values of these columns reveal that, the proposed algorithm can obtain good solutions in reasonably short computational time.

Table 1 Computational results for $r=25$

m	n	Average Run-Time (CPU Seconds)	Average Run-Time for Obtaining the Best Solution (CPU Seconds)	Average Best <i>Ratio</i>
100	5	228	139	0.95
100	10	322	214	0.96
150	5	328	130	0.95
150	10	481	236	0.96
200	5	435	321	0.95
200	10	674	482	0.96
250	5	576	409	0.96
250	10	894	425	0.96

Table 2 Computational results for $r=50$

m	n	Average Run-Time (CPU Seconds)	Average Run-Time for Obtaining the Best Solution (CPU Seconds)	Average Best <i>Ratio</i>
100	5	659	295	0.90
100	10	884	352	0.91
150	5	974	690	0.90
150	10	1238	724	0.92
200	5	1189	439	0.91
200	10	1659	1145	0.91
250	5	1561	1114	0.90
250	10	2120	1399	0.91

6. CONCLUSIONS

In this paper, the problem of allocating the advertising time on prime-time to advertisers with the objective of maximizing the revenue of a TV channel is considered. We formulated the problem in the form of a binary integer programming model. The proposed mathematical model is based on the well known procedure of combinatorial auction mechanism. Due to the combinatorial nature of this problem, the computational time for obtaining the optimal solution grows exponentially as the number variables increases (this problem is reported as an NP-Complete problem). This prohibits the use of exact solution approach for solving the real-sized problems. This deficiency can be overcome by the use of a heuristic approach. We therefore developed a steady-state genetic algorithm to find a good or probably near-optimal solution for the problem.

We then evaluated the computational time for obtaining solutions for the test problems as well as the quality of solutions through a computational study. To conduct this computational study, we generated a set of test problems with different sizes. Totally 16 classes of test problems with different sizes was randomly generated. For each class of the test problem we generated 10 different test problems. We then solved the generated test problems using our developed GA and also solved the LP relaxation of the same test problems. Considering the solution of LP-relaxed problem as an upper bound of the optimal solution for each test problem, we evaluated the quality of the GA solutions.

The results obtained by the computational study, reveal that in average, GA's solutions deviate less than 10 percent of the upper bound of the optimal values. The results demonstrate that the proposed genetic algorithm is capable of obtaining high-quality solutions for the problem.

REFERENCES

- [1] Benoist T., Bourreau E., Rottembourg B. (2007), The TV-break packing problem; *European Journal of Operational Research* 176; 1371-1386.
- [2] Bichler M., Davenport A., Hohner G., Kalagnanam J., In Cramton P., Shoham Y., Steinberg R. (2006), Combinatorial Auctions, Chapter 22; Industrial Procurement Auctions, MIT Press.
- [3] Bollapragada S., Cheng H., Philips M., Garbiras M., Scholes M., Gibbs T., Humhreville M. (2002), NBCs optimization systems increase revenues and productivity; *Interfaces* 32(1); 47-60.
- [4] Bollapragada S., Garbiras M. (2004), Scheduling commercials on broadcast television; *Operations Research* 52(3); 337-345.
- [5] Bollapragada S., Bussieck M., Mallik S. (2004), Scheduling commercial videotapes in broadcast television; *Operations Research* 52(5); 679-689.
- [6] Brown A.R. (1969), Selling television time: An optimization problem; *Computer Journal* 12; 201-206.
- [7] Brusco M.J. (2008), Scheduling advertising slots for television; *Journal of the Operational Research Society* 59(10); 1373-1382.
- [8] Cramton P., Shoham Y., Steinberg R. (2005), Combinatorial Auctions, MIT Press.
- [9] Holland J. (1975), *Adaptation in Natural and Artificial Systems*; University of Michigan Press.
- [10] Jones J.J. (2000), Incompletely Specified Combinatorial Auction: An Alternative Allocation Mechanism for Business to Business Negotiations; PhD Dissertation, University of Florida, FL.
- [11] Kimms A., Muller-Bungart M. (2007), Revenue management for broadcasting commercials; *International Journal of Revenue Management* 1; 28-44.
- [12] Leyard J.O., Olson M., Porter D., Swanson J.A., Torma D.P. (2002), The first use of combined-value auction for transportation services; *Interfaces* 32(5); 4-12.
- [13] Mao J., Shi J., Wanitwattanakosol J., Watanabe Sh. (2011), An ACO-based algorithm for optimizing the revenue of TV advertisement using credit information; *International Journal of Revenue Management* 5(2); 109-120.
- [14] Mihiotis A., Tsakiris I. (2004), A mathematical programming study of advertising allocation problem; *Applied Mathematics and Computation* 148; 373-379.
- [15] Pereira P.A., Fontes F.A.C.C., Fontes D.B.M.M. (2007), A decision support system for planning promotion time slots; *Operations Research Proceedings*; 147-152.
- [16] Rassenti S., Smith V., Bulfin R. (1982), A combinatorial auction mechanism for airport time slot allocation; *Bell Journal of Economics* 13(2); 402-417.

- [17] Sandholm T., Suri S., Gilpin A., Levine D. (2002), Winner determination in combinatorial auction generalizations; *Proceeding of Autonomous Agents and Multi-Agent Systems Conference*; 69-76.
- [18] Wuang M.S., Yang C.L., Huang R.H., Chuang S.P. (2010), Scheduling of television commercials; *IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*; Macao, 803-807.
- [19] Zhang X. (2006), Mathematical models for the television advertising allocation problem; *International Journal of Operational Research* 1(3); 302-322.