Searching Optimal Buyer Coalition Structure by Ant Colony Optimization

Anon Sukstrienwong

Abstract— In recent years, several buyer coalition schemes have been proposed by researchers in order to form effective coalitions and achieve the maximum benefit for consumers in an electronic market. However, there are few algorithms applying the ant colony optimization for forming buyer coalition. In this paper, we present the approach based on the Ant Colony Optimization (ACO). The approach called the Ant Colony Optimization for Forming of Buyer Coalition (ACO_FBC) algorithm for the formation of buyer coalition with bundles of items. The algorithm involves searching for the optimal buyer coalition structure by partitioning the whole group of buyers into smaller coalitions so that the aggregate of discount of the whole buyers is maximized. A number of artificial ants search to find the best disjoint subgroups of all buyers based on the total utility functions. The results of the ACO_FBC simulation are compared with the genetic algorithm (GAs) in the terms of the global optimal buyers' benefits. It indicates that in most situations our proposed algorithm significantly improves the utility of the buyer coalition.

Keywords— Ant colony optimization, buyer coalition, coalition structure, electronic commerce, simulation.

I. INTRODUCTION

To date, there exist several online shops available on the Internet such as, http://www.alibaba.com, http://www.amazon.com, and http://www.staples.com. Some of these online shops adopt many strategies to expedite their selling. Bundle of items can be an important issue for sellers. Bundle of items are packed with a variety of items and priced a few lower than what they would be if bought individually. Some bundles give bonus items that can be obtained by buying a bundle on the Internet. In addition, some online shops like http://www.aliexpress.com offer wholesale price to customers. These prices are usually about half the price of something that could be purchased at retail store. But, the wholesale sellers offer a price at a 100% to the retail customer. On the other side, buyers prefer to obtain a deduction from the price list offered by sellers in return for payment. The accessibility of the Internet and lower costs of doing transactions have given rise in customers bargaining power and intense global competition [28]. Of course, bargaining is one of the traditional strategies for buyers and seller to reach beneficial agreements. One common shopping tactic which most buyers are likely to make is a group buying because a large group of buyers gains more negotiating power. A buyer coalition is set of buyers who agree to join together to bargain with sellers, so buyers can gain volume discount prices. The other strategy is the buyer coalition scheme.

Several buyer coalition schemes exist with the aim of having the best group utility [1], [3], [4]. However, few schemes consider forming group of buyer with bundles of items which can be often occurring in the real world. There are several opportunities that it can be happened, such as a case that buyers cannot purchase the bundles of items by their own because the packages of products sold by sellers are composed of hundreds of items or multiple type of items.

The algorithm in [5] called GroupPackageString scheme applies genetic algorithms (GAs) to form buyer coalitions with bundle of items. However, this algorithm does not consider the situation of partitioning the whole group of buyers into smaller groups. The partitioned group is called a coalition structure (CS). Some researchers have developed and evaluated the performance of anytime CSG algorithms to search for optimal coalition structures in characteristic function games (CFGs) [6]. It is also applied in many complex autonomous applications as electronic marketplace, [7] –[11]. The CS aims to maximize the utility of the coalitions, but often the number of coalition structures is too large to allow for the exhaustive search for the optimal one [2]. The optimal solution of the problems can result at any of the n levels of the coalition structure. Furthermore, finding optimal coalition structure is NP-complete. The size of the search space is exponential in the number of agents.

Given a set of m members, $A = \{a_1, a_2, \ldots, a_m\}$ and a subset or coalition $C \subseteq A$, there are two challenging stages involving in this paper:

- Search for the best coalition structure of all in which the union of subsets equals A by using the ant colony algorithm (ACO).
- Compute the total utility of the whole group.

Given a coalition structure CS, we denote the value of CS by

$$V(CS) = \sum_{c \in CS} v(C, CS), \tag{1}$$

where $v(C, CS)$ is the value of coalition structure of $C \in CS$. And, the optimal coalition structure is noted as

Anon Sukstrienwong is with School of Science and Technology, Information and Technology Department, Bangkok University, Bangkok, 10110, Thailand. (e-mail: anon.sur@bu.ac.th)

$$CS^* = \arg \max_{CS \in L} V(CS). \qquad (2)$$

Hence, L_i is known as a layer, and the set of all CSs is $L = \bigcap_{i=1}^{n} L_i$. In CFG the value of each coalition is given by a characteristic function which is simply defined as the sum of the values of the coalitions that it contains. Reference [12] shows the calculation of the total number of coalition structures as follow,

$$\sum_{i=1}^{a} Z(a,i), \qquad (3)$$

where a is the number of agents. And $Z(a,i)$ is the number of coalition structures with i coalitions,

$$Z(a,i) = i * Z(a-1,i) + Z(a-1,i-1), \qquad (4)$$

where $Z(a,a) = Z(a,1) = 1$.

If a is small, a = 4, the number of coalition structures is 15, see Fig. 1. However, when number of agents increases linearly, the size of the problem, 2^n, increases exponentially [27]. This is such a difficult task to search for optimal solution when the number of agent is bigger. So, the algorithm called ACO_FBC is proposed to search for the optimal solution. The proposed algorithm is based on ant colony algorithms (ACO) which are inspired from the behavior of real ants. The major shortcoming of the ACO_FBC is that it provides no guarantee to find the optimal solution because ACO is one of the heuristic functions. However, it seems to work well in our practice.

The purpose of this paper is to search for the optimal buyer coalition structure by applying ACO technique. The paper is divided into five sections including this introduction. The rest of the paper is organized as follows. Section 2 describes basic ant colony optimization. In section 3, we show the motivated example including the mathematical details of the proposed algorithm. In section 4, we show the experiments. To ensure the quality of the algorithm, the simulation results are compared with the GroupPackageString scheme. Finally, the conclusions and future work are in the last section.

II. Ant colony optimization Background

Ant colony optimization (ACO) is a probabilistic technique for finding optimal paths in fully connected graphs through a guided search, by making use of the pheromone information [29]. It is a paradigm for designing metaheuristic algorithm for combinatorial optimization problems [15]. The main idea in ant colony algorithms is to use artificial ants that iteratively construct solutions to combinatorial optimization problems. The first ACO algorithm was initially proposed by Colorni, Dorigo and Maniezzo [21] – [23] in 1997 which known as Ant System (AS). Now, there are several adaptations of such algorithms to complex optimization problems [9], [17]–[19], [24]. The "global" ants perform a simple evaluation of some regions defined in the search space, in order to update the regions fitness. The ACO was applied to several problems such as the traveling salesman problem [17] and the shop scheduling problem and mixed shop scheduling [18]. In nature, real ants are capable of finding the shortest path from a food source to their nest without using visual cues [20].

The general structure of ACO algorithms can be described as follows.

Step 1: Initialize the pheromone trails and parameters.

Step 2: While (termination condition is not met)

do the following:

- Construct a solution;
- Improve the solution by local search;
- Update the pheromone trail or trail intensities.

Step 3: Return the best solution found.

In the ACO, the process begins by initiating m completely random ants. Theses artificial ants build solutions to an optimization problem while updating pheromone information on its visited tail. Artificial ants build a feasible solution by recurrently applying a stochastic greedy rule. While creating its tour, an ant deposits a substance called pheromone on the ground and follows the path by previously pheromone deposited by other ants. Once all ants have completed their tours, the ant which found the best solution deposits the amount of pheromone on the tour according to the pheromone trail update rule. The best solution found so far in the current iteration is used to update the pheromone information. The pheromone τ_{ij}, associated with the line joining i and j, is updated as follow:

$$\tau_{ij} \leftarrow (1-\rho)\cdot \tau_{ij} + \sum_{k=1}^{m} \Delta\tau_{ij}^{k}, \qquad (5)$$

where ρ is the evaporation rate which $\rho \in (0,1]$. The reason for this is that old pheromone should not have too strong an influence on the future. And $\Delta\tau_{ij}^{k}$ is the amount of pheromone laid on line (i, j) by ant k:

$$\Delta\tau_{ij}^{k} = \begin{cases} Q/L_k & \text{If line (i, j) is used by ant k.} \\ 0 & \text{otherwise,} \end{cases} \qquad (6)$$

where Q is a constant, and L_k is the length of the tour performed by ant k. By constructing a solution, it starts from the starting city to visit an unvisited city. When being at the city i, the ant k selects the city j to visit through a stochastic mechanism with a probability p_{ij}^{k} given by:

$$p_{ij}^{k}=\begin{cases}\dfrac{\tau_{ij}^{\alpha}\cdot\eta_{ij}^{\beta}}{\sum_{c_{il}\in N_{i}^{k}}\tau_{il}^{\alpha}\cdot\eta_{il}^{\beta}} & \text{if } j\in N_{k}^{i}\\ 0 & \text{otherwise,}\end{cases} \tag{7}$$

where N_k^i is a set of feasible neighborhood of ant k, representing the set of cities what ant k has not been visited. α and β are two parameters which determine the relative influence of pheromone trail and heuristic information, and η_{ij} , which is given by:

$$\eta_{ij}=\frac{1}{d_{ij}}, \tag{8}$$

where d_{ij} is the length of the tour performed by ant k between cities i and j.

III. Searching Optimal Buyer Coalition by Ants

This section gives the formal definition of the buyer coalition addressed in this paper. The motivated example of our problem is shown to demonstrate the difficulty of the problem.

A. The motivated example

Online shopping information such as new product release, promotion, and other news are mostly retrieved manually by prospective buyers from individual website or webpage [25]. Suppose some sellers sell three kinds of product, x_1, x_2 and x_3. Price is one of the most influential factors that can easily increase or decrease product demand [26]. Traditionally, these sellers prepare a large stock of goods with many attractive prices shown in Table 1. We assume that these sellers can supply unlimited items of any products. And, seller policy is based on on the number of items. The more items is with a single package, the more discount. The package number 1–3 are single-item packages. The package number 4 of seller 1 composed of one item of X_1, X_2 and X_3 is set to be sold at the price of 54.0 dollars, which is 10% of the original price. The package number 4 of the same seller comprised of more items (10 items of X_2 and 1 item of X_3) is set to be sold at the price of 180 which is 20% discount of the original price. The maximum discount of the seller number 1 is 30%. The discount policy for the seller number 2 is different for the seller number 1. The seller number 2 has made 5 different packages. The package number 1 – 3 are also a single-item package. The package number 4 of the seller number 2 is comprised of 50 items of X_1. It is set to be sold at the price of 525.0 which is about 25% discount of the original price.

TABLE I. The Example of Price List

Seller	Product (k=3)					Discount of original price (%)
	Package	X_1	X_2	X_3	Price ($)	
1	1	1	0	0	15.0	-
	3	0	1	0	20.0	-
	2	0	0	1	25.0	-
	3	1	1	1	54.0	10%
	4	0	10	1	180.0	20%
	5	10	0	2	170.0	15%
2	1	1	0	24	430.5	30%
	2	1	0	0	14.0	-
	3	0	1	0	22.0	-
	4	0	0	1	27.0	-
	5	50	0	0	525.0	25%
	6	0	50	0	770.0	30%

'0' means that the sellers do not put the item in the package.

TABLE II. The Example of Buyer Requests

Buyer	Buyer requests (k=3)			Locations		
	X_1	X_2	X_3	L_1=10	L_2=15	L_3=30
a_1	5	0	0	√	0	0
a_2	5	0	2	√	0	0
a_3	0	40	0	0	0	√
a_4	0	10	0	0	√	0

'0' means that the buyers do not want to buy that item.

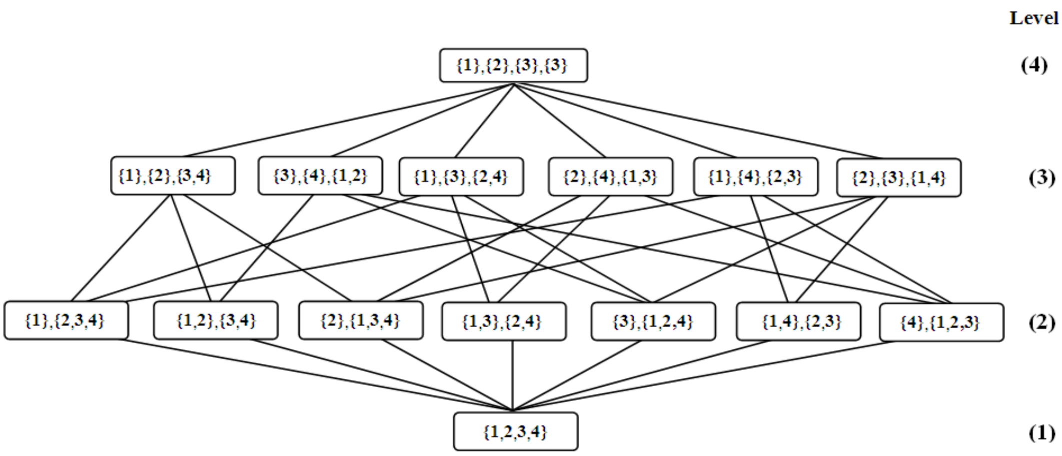
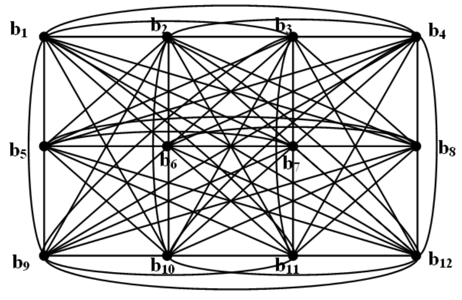


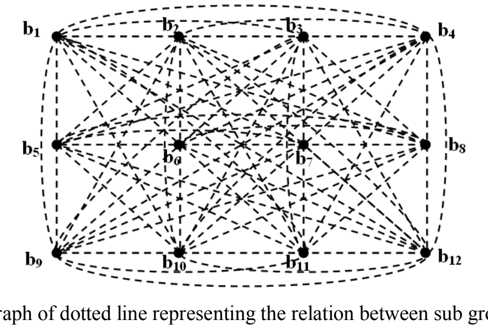
Fig. 1 Coalition structure of 4 buyers shown in [4]

Moreover, if the buyer purchases more items from one seller, this buyer may get free shipping. In electronic marketplaces, many buyers come from different places because the buyers order products from anywhere from the internet. Suppose there are three buyers called a_1, a_2, a_3, and a_4. After they have seen the price list of both sellers, they have made their decision to buy some products. Of course, buyers prefer to purchase products as lower as they can. However, they do not want to buy the whole package to get the special price. They have come to join their requests in the group buying. Their requests are shown in the Table 2.

As we can see, the a_1 and a_2 are resided in the same area, location L_1. If they join their request, the seller sent the whole package to one of them without shipping cost. And, the best package for both a_1 and a_2 is package number 5 from the seller number 1. They would pay at most 170.0 dollars. Suppose a_3 and a_4 assemble their request to buy the package number 5 from the seller number 2. They need to pay at least 525 dollars including the shipping cost. It is because they are resided in the different areas. In general case, the seller would send the whole package to one person which has the largest demand without the shipping cost. So, the seller number 2 sends the package number 5 to a_3. Then, a_3 sends 10 items of X_2 to a_4 with the cost of 15. The total spending for both a_3 and a_4 are 770+15 = 785 dollars. As the buyer coalition is formed,

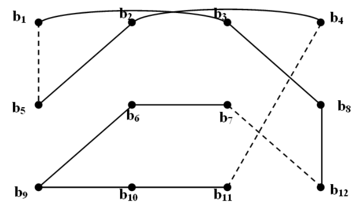


a) Graph of solid line representing the membership of the same sub group

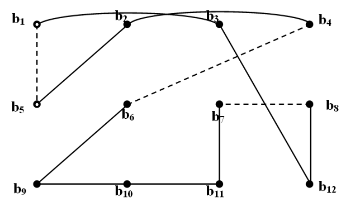


b) Graph of dotted line representing the relation between sub groups

Fig.2 Representing the possible relationship between buyers (number of buyers = 12)



a) Graph of solid line representing the membership of the same sub group



b) Graph of dotted line representing the relation between sub groups

Fig.3 The example of graph representing for $\{\{b_1,b_3,b_8,b_{12}\},\{b_2,b_4,b_5\},\{b_6,b_7,b_9,b_{10},b_{11}\}\}$

the total spend is 170+785 = 955 dollars. If all buyers unite into only one group, they need to buy a set of package number 5 from the seller number 1 and a set of the package number 4 from the seller number 2. The total cost is 170+770 = 940 dollars. However, buyers are located in different places. The seller sent all products to a_3 because a_3 has the biggest order. When a_3 gets the products, a_3 passes 5 items of X_1 to a_1 with the shipping cost of 10. And, a_2 sends 5 items of X_1 and 2 items of X_3 to a_2 with the shipping cost of 10 dollars. Finally, a_2 sends 10 items of X_2 to a_4 with the shipping cost of 30 dollars. The total shipping cost is 10+10+15 = 35. So, the total spending from forming the group buying is 940+35 = 975 dollars. As we can see that in this case forming the group buying by partitioned the whole group into smaller groups, $\{a_1, a_2\}$ and $\{a_3, a_4\}$, uses lower cost than forming the whole buyers into one big group $\{a_1, a_2, a_3, a_4\}$, which is about 975-955 = 20 dollars. The problem is that how can we find the optimal solution. Specially, when the number of buyers is big, the possible number of partitioned groups is also big.

B. The creation of paths through the disjoint subsets of all buyers

In the first step, the problem is represented as graph where the optimum subgroup of buyers can be defined in a certain way through this graph. To follow the ants to walk through the graph of CS, the new representation of graph is defined as shown in Fig. 2. Given a set of 12 buyers, $B = \{b_1, b_2,\ldots, b_{12}\}$, there are two possible lines connecting between each buyers. However, due to the several lines between buyers, the graph is split into two graphs; one is the solid line representing the membership of the same sub group, and the other is dotted line representing the relation between sub groups. For n of buyers, the total lines connected between two buyers is equals 2(n-1). And, the total lines of the problem is equal to 2(n(n+1)/2) = n(n+1). So, for n = 12, there are 2(12-1) = 22 lines between two buyers and the total line of the whole graph is 12(12+1) = 156. If one of the coalition structure is $\{\{b_1,b_3,b_8,b_{12}\},\{b_2,b_4,b_5\},\{b_6,b_7,b_9,b_{10},b_{11}\}\}$, the possible graphs of this particular coalition structure problem is shown in Fig. 3. There are 3 sub coalitions; $\{b_1,b_3,b_8,b_{12}\}$, $\{b_2,b_4,b_5\}$ and $\{b_6,b_7,b_9,b_{10},b_{11}\}$. For the first sub coalition, the solid line connects two members in $\{b_1,b_3,b_8,b_{12}\}$.

Rule applied to the formation of buyer coalition with n of buyers.

a. There are two types of lines; solid line and dotted line; connected between two buyers.
b. For generation the solution via ACO_FBC algorithm, the algorithm allows each buyer holds exactly two lines.

C. *Problem formulization*

The proposed algorithm involves partitioning a set of elements into subsets based on the utility function that are associated to each subset. The formulization of the ACO_FBC algorithm can be described as below.

Given a set of buyers A = $\{a_1, a_2, \ldots, a_m\}$, there are two kinds of relationships between two buyers which are represented by edges. If buyer a_i and a_j, i≠j, are in the same subgroup, there is a path using solid lines to walk from a_i and a_j. But, it is not necessary to have a solid line directly connecting between a_i and a_j. Let A is divided into n different groups $(C_1,C_2,\ldots,C_n)$ and $\prod_{k=1}^{n} C_k = \phi, \bigcup_{k=1}^{n} C_k = A$, where 1≤k≤m. There exits a dotted line connecting between C_k and C_l where k ≠ l. For example, let a set of A = {1, 2, 3, 4, 5, 6}, the coalition structure of A are shown in Fig. 4. But, for our method the graph can be represented in Fig. 5 (a). Buyers are represented as vertices. So, there are exactly 2*(6-1) = 10 lines for each buyer to connect to others. If A is divided into two subgroups, C_1 = {1, 2, 4} and C_2 = {3, 5, 6}, the graph represented the relation among buyers can be shown in Fig. 5(b). Also, it can be represented as Fig 5(c). These graphs are created during the search by artificial ants so they are called the ACO_FBC graph. If A is divided into three subgroups, C_1 = {1, 2, 4}, C_2 = {3, 6}, and C_3 = {5}, then the graph represented the relation among buyers can be shown in Fig. 5(d). The example of the ACO_FBC graph of {1,2,3}{4}{5}{6} is shown in Fig. 5(e).

The ants built the path from each buyer to unvisited buyers until all the buyers have been visited. This means that each buyer can be visited only one time during the constructing of the path except for the first buyer. Then, the ACO_FBC graph becomes the closed graph. In addition, the total number of edges for constructing the ACO_FBC graph is m-1, where m is the number of buyers. In this paper, the proposed algorithm relies on the assumption that the value of a coalition is independent of other coalitions in the coalition structure. All buyers in the group participate in the process of the algorithm, and each buyer is represented exactly once in the ACO_FBC graph. At the beginning all of the pheromone values of each package are initialized to the very small value c, 0 < c ≤ 1. The artificial ant, called ant m, chooses all members for finding the best group's utility on return. After initializing the problem graph with a small amount of pheromones and defining each ant's starting point, several ants run for a certain number of iterations. The probability of the ant m to choose one member called i to join with the other called j with the relation k (where k ∈T = {dotted line, solid line}) is $p_{ij_k}^{m}$ defined formally as below:

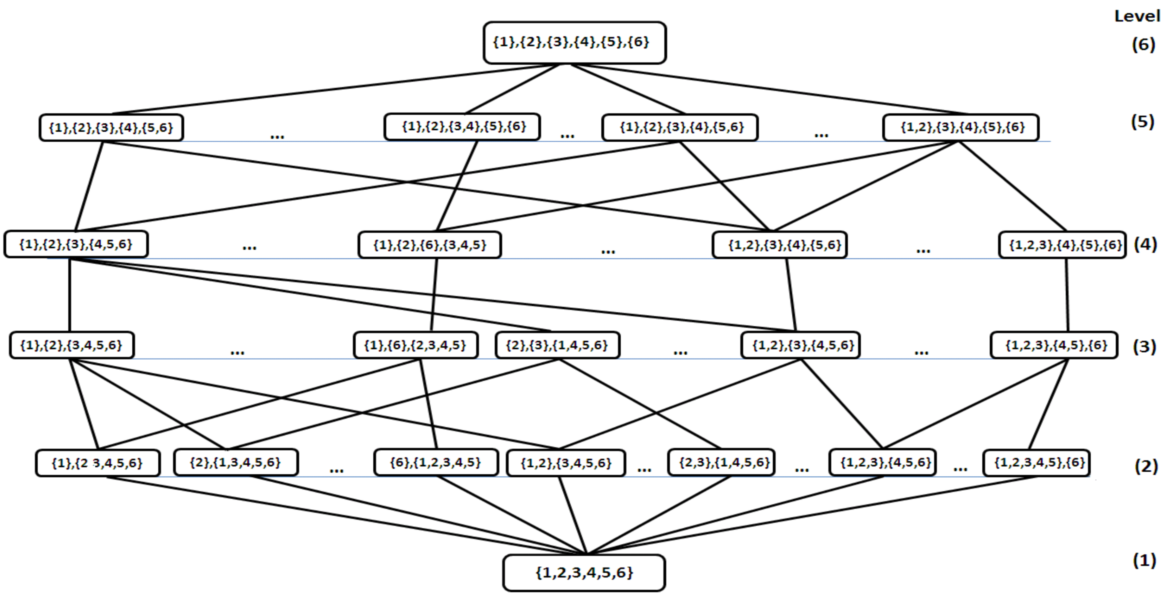
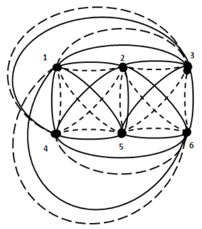
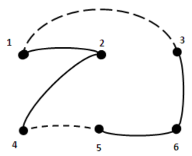


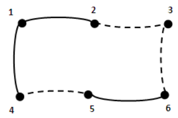
Fig. 4 Coalition structure of six buyers



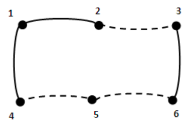
(a) All of the possible paths connection among six buyers



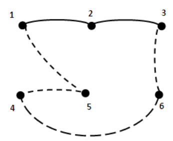
(b) {1, 2, 4} {3, 5, 6}



(c) {1, 2, 4} {3, 5, 6}



(d) {1, 2, 4} {3, 6} {5}



(e) {1, 2, 3} {4} {5} {6}

Fig.5 The ACO_CS graph of {1,2,6} {3,4} {5}

$$p^m_{ij_k} = \begin{cases} \dfrac{\left(\tau^m_{ij_k}\right)^\alpha \left(\eta_{ij_k}\right)^\beta}{\sum_i \sum_{l \in A} \sum_{d \in T} \left(\tau^m_{il_d}\right)^\alpha \left(\eta_{il_d}\right)^\beta} & \text{if } l \in \text{A and } a_l \text{ has not been selected} \\ 0 & \text{otherwise,} \end{cases} \quad (9)$$

where α and β are two parameters which determine the relative influence of pheromone trail and heuristic information and $\Delta\tau^m_{ij_k}$ is the amount of pheromone laid on the line between a_i and a_j on either solid line or dotted line by the ant m defined as follow:

$$\Delta\tau^m_{ij_k} \begin{cases} 1/(U_{c_n}) & \text{if } a_i \text{ and } a_j \text{ were selected by ant m with the relation k,} \\ 0 & \text{otherwise.} \end{cases} \quad (10)$$

If k = dotted line, then $C_n = \{\{a_i\}\{a_j\}\}$. If k = solid line, $C_n = \{\{a_i a_j\}\}$. And, η_{ij_k} is given by:

$$\eta_{ij_k} = \begin{cases} 1/(D - U_{c_n = \{\{a_i\}\{a_j\}\}}) & \text{if k = 0} \\ 1/(D - U_{c_n = \{\{a_i, a_j\}\}}) & \text{otherwise,} \end{cases} \quad (11)$$

where D is the constant value, and both $U_{\{\{a_i\}\{a_j\}\}}$ and $U_{\{\{a_i, a_j\}\}}$ are derived by (8) and (9). The pheromone τ_{ij}, associated with the line joining a_i and a_j, is updated as follow:

$$\tau_{ij} \leftarrow (1-\rho) \cdot \tau_{ij} + \sum_{k=1}^{m} \Delta\tau^m_{ij_k}, \quad (12)$$

D. ACO_FBC algorithm

The ACO_FBC algorithm for forming buyer group with bundles of items can be described by the following procedure:

Procedure ACO_FBC(){
Initialization all pheromone values to a small numerical constant c > 0
- Initialization of the ACO_FBC
- T = {0 = dotted line, 1 = solid line};
while not (isFinish(Iteration)){
for Ant = 1 to MaxAnt {
ManageAntsActivity();
EvaporatePheromone();
Calculate the Utility based on (8) and (9) and save the best solution found so far.
UpdatePheromone();
}
}
}

ManageAntsActivity(){
While not (isAntFinish(tour)){
Choose a buyer a_i to be visited with probability $p^m_{ij_k}$ in (9), (10), and (11).
If selected path is a dotted line (T=0), then $\{a_i\}$ separate with $\{a_j\}$.
If selected path is a solid line (T=1), then $\{a_i\}$ union with $\{a_j\}$.
}
}

EvaporatePheromone(){
Old pheromone should not have too strong an influence on the future. The evaporation rate value is ρ which is initialized to be small, $\rho \in (0,1]$.
}

UpdatePheromone(){
Update the all the path according to (12).
}

E. Algorithm revisited and its Example

Given a set of six agents as shown in Table 1, the system as a whole must seek a maximization of its benefits. The problem we solve in this paper depends on two factors, buyer request and buyer location. Each agent can have several requests of X_i. Agents join with others when they get the best utility that is defined in the utility function (see (13)).

Suppose the current ant, called ant m, chooses the agent 1 for the starting point. Then, the ant m searches for the next agents based on the probability $p_{ij_k}^{m}$ shown in (9). If three agents, 1, 2 and 6, have been chosen respectively to be in the same sub group denoted as {1, 2, 6}. In the ACO_CS algorithm, the current ant generates the path connecting among selected agents. Therefore, solid lines are used to join three of these agents, See Fig. 6(a). The utility of the subgroup {1, 2, 6} is 575, see the calculation of $\text{Util}_{\{1,2,6\}}$. Again, the current ant finds the next agent with the probability shown in (8). Suppose that the ant m chooses the path from vertex 6 via a dotted line to agent 3, see Fig. 6(b). If the agent number 3 and 4 have high possibility to be chosen into the same sub coalition, denoted as {3, 4}, then the ant chooses a solid line between 3 and 4, see Fig. 6(c). The utility of {3, 4} is 450, see the calculation of $\text{Util}_{\{3,4\}}$. The last agent, agent number 5, is joined with no one. It is an isolated agent, therefore this agent is connected with the other by two dotted lines, see Fig. 6(d) and Fig. 6(e)). Its utility is 330, see the calculation of $\text{Util}_{\{5\}}$. Then, the coalition can be written as {1,2,6}{3,4}{5} with the grand total utility of 1355. Finally, the ant m deposits the amount of pheromone on the trail according to the pheromone value in (10).

$$U_{C_n} = \sum_{a_m \in C_n} (K_1 * \sum_{r=1}^{k} X_r^{a_m}) + K_2 * Max\begin{pmatrix} \sum_{a_m \in C_n} (L_{i=1}^{a_m} * \sum_{r=1}^{k} X_r^{a_m}), \\ \sum_{a_m \in C_n} (L_{i=2}^{a_m} * \sum_{r=1}^{k} X_r^{a_m}), \\ \dots \\ \sum_{a_m \in C_n} (L_{i=z}^{a_m} * \sum_{r=1}^{k} X_r^{a_m}), \end{pmatrix} - \sum_{a_m \in C_n} (\sum_{i=1}^{z} (L_i^{a_m} * \sum_{r=1}^{k} X_r^{a_m})) - K_3, \tag{13}$$

where K_1, K_2 and K_3 are the constant, and $X_r^{a_m}$ is the request X_r of an agent a_m. So,

$$v(CS) = \sum_{C_n} U_{C_n} \cdot \tag{14}$$

Let's the constant K_1=100, K_2 = 2, and K_3 =150. So, utility of each sub coalition, {1, 2, 6}, {3, 4}, and {5} can be calculated as follows.

$\text{Util}_{\{1,2,6\}}$=K_1(2+1+2+2)+K_2(4*15)-(4*15+2*10+1*15)-K_3

= 100*7+2*60-95-150

= 575

$\text{Util}_{\{3,4\}}$=K_1(2+1+3)+K_2(2*30)-(2*30+4*15)- K_3

= 100*6+2*60-120-150

= 450

$\text{Util}_{\{5\}}$=K_1*4+K_2(4*10)-K_3

= 100*4+2*40-150

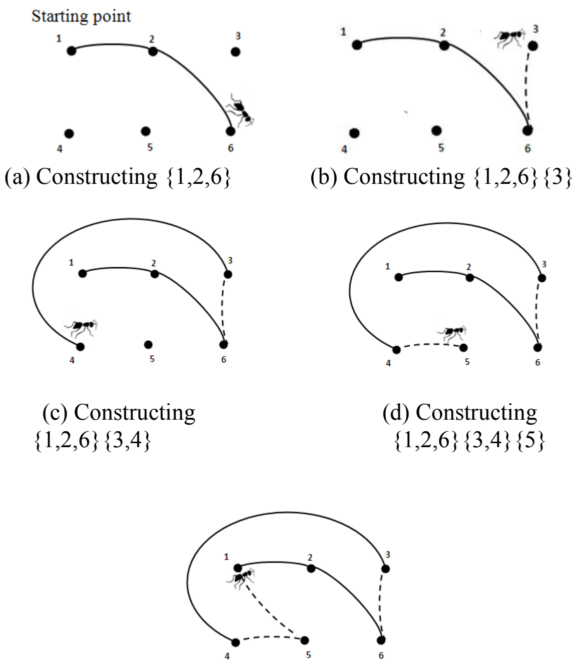
= 330

$$v(CS) = \sum_{C_n} U_{C_n}$$

= $\text{Util}_{\{1,2,6\}}$+ $\text{Util}_{\{3,4\}}$+ $\text{Util}_{\{5\}}$

= 575+450+330

= 1355



(a) Constructing {1,2,6}

(b) Constructing {1,2,6}{3}

(c) Constructing {1,2,6}{3,4}

(d) Constructing {1,2,6}{3,4}{5}

(e) The final graph of {1,2,6}{3,4}{5}

Fig. 6 The ACO_CS graph of {1,2,6}{3,4}{5}

IV. Simulation and Experiments

This section explains the idea of how the ACO_CS algorithm by using an empirical example. Then, it shows the experiment results in detail. In the experiment, we use the parameterized function as stated in (13) and (14), so that we can regulate some data and alter the search space to observe our proposed function in practice. In our algorithm, we assume that all buyers are requested to participate in constructing the coalition structure. To test the ACO_FBC algorithm, a simulation was developed using Java programming language. The simulation runs on a Pentium(R) D CPU 2.80 GHz, 2 GB of RAM, IBM PC. ACO_FBC parameters include α=0.5, β=1, and the number of artificial ants = 1000. Several experiments have been conducted using a different set of buyers by random with 5, 10, 20, and 30 buyers respectively while the price list is shown in Table 1. The summary results of our experiments are presented in

Table 3. In most cases, the average result (ten runs) of all tests derived by the ACO_FBC algorithm is better than the result of GroupPackageString scheme. For test 1 we can see that when the number of buyers is small, n=5, both ACO_FBC algorithm and GroupPackageString scheme give similar results. This is because for this specific test the whole group of buyers cannot be partitioned. All of these five buyers should be joining in a single group, so both ACO_FBC algorithm and GroupPackageString scheme have very similar results. However, when the number of buyers is bigger (n > 5), ACO_FBC algorithm seems to work better, see test number 2-4. It searches the better coalition structure yielding the optimal score. In average, ACO_FBC algorithm yields about (8.37+16.52+10.64)/3 = 11.84% better than the GroupPackageString scheme. This is because the GroupPackageString scheme searches for the optimal result for the whole group of buyers, while ACO_FBC algorithm partitions the whole buyers into smaller subgroups in order to find the best utility.

TABLE III. SIMULATION RESULTS

Test No.	Number of buyers (n)	**Experimental Results Average of Utility (10 runs)**			
		ACO_FBC algorithm		GroupPackageString	
		Number of sub coalitions	Average Utility (% over GroupPackageString)	Number of sub coalitions	Average Utility
1	5	2	226.0 (0%)	1	226.0
2	10	2	492.0 (8.37%)	1	454.0
3	20	4	818.0 (16.52%)	1	702.0
4	30	5	1456.0 (10.64%)	1	1316.0

V. CONCLUSIONS AND FUTURE WORK

In this paper, we presented a method searching for the optimal buyer coalition structure where the buyer coalition is formed through the use of ant colony optimization technique. The proposed algorithm called ACO_FBC algorithm is based on an imitation of the foraging behavior of real ants. The efficiency of the ACO_FBC algorithm was evaluated through the performance of several experiments, four different sets of buyers by complete random. The simulation results show that the average utility of any coalitions formed by ACO_FBC is better than the GroupPackageString scheme. It has been concluded that the ACO_FBC algorithm can be efficiently used for searching the optimal buyer coalition structure problems. However, there are some restrictive assumptions for our proposed algorithm as follow: 1) The buyer coalition is formed concerning only the price attribute. 2) Buyers can make order requests with several choices of items. 3) Bundle of items is packed with a variety of items in a package at one price which is below the sum of the independent prices. The average price of each item will be cheaper than the price of a single-item package. 4) Sellers can supply unlimited items of any products 5) the discount policy of sellers based on the number of items bundled in the package. These restrictions can be extended to investigate in future research. We also plan to adapt the proposed algorithm to other real-complex world problems to see how well to apply. Future work will include investigation of the ACO_FBC algorithm performance in other algorithms and real life problems.

REFERENCES

[1] K. Larson and T. W. Sandholm, "Anytime coalition structure generation: An average case study," In Proceedings of the Third International Conference on Autonomous Agents, New York: NY, ACM Press, pp. 40–47, 1999.

[2] T.W. Sandholm, K. Larson, M. Andersson, O. Shehory, and F. Tohme , "Anytime coalition structure generation with worst case guarantees," in: *Proc. of the Fifteent National Conference on Artificial Intelligence*, Menlo Park, CA, AAAI Press/MIT Press, pp. 46–53, 1998.

[3] M. Tsvetovat and K. Sycara, "Customer coalitions in the electronic marketplace," In *Proc. of the fourth International Conference on Autonomous Agents*, pp. 263-264, 2000.

[4] T. Rahwan, S. D. Ramchurn, V. D. Dang, and N. R. Jennings, "Near-optimal anytime coalition structure generation," In: *IJCAI*, 2007, pp. 2365–2371.

[5] D. V. Dang and N. R. Jennings (2004), "Generating coalition structures with finite bound from the optimal guarantees," In AAMAS, pp. 564–571.

[6] T. Michalak, A. Dowell, P. McBurney, and M. Wooldridge (2008). Optimal coalition structure generation in partition function games. In Proceedings of the 18th European Conference on Artificial Intelligence (ECAI-08), pp. 388–392.

[7] M. Fasli, "Agent technology for E-commerce," London: Wiley, 2007.

[8] M. Klusch, "Agent-mediated trading: intelligent agents and e-business," in: A.L.G. Hayzelden, R. Bourne (Eds.), *Agent Technology Applied to Networked Systems*, Wiley, Chichester, 2000.

[9] A. Sukstrienwong (2010), "Ant colony optimization for buyer coalition with bundle of items," in: *Proc. of the World Congress on Engineering 2010*, Volume 1. WCE 2010, June 30 - July 2, 2010, London, U.K, pp. 38-43.

[10] Laor, B., Leung, H. F., Boonjing, V., Dickson, K. W., "Forming Buyer Coalitions with Bundles of Items". In: *Nguyen NT, Hakansson A, Hartung R., Howlett R., Jain LC (eds.)*, KES-AMSTA 2009. LNAI 5559-0717, pp. 121-138. Springer, Heidelberg.

[11] Boongasame, L. and Sukstrienwong, A., "Buyer coalitions with bundles of items by using genetic algorithm," in: *Emerging Intelligent Computing Technology and Applications*, Springer, Heidelberg, pp. 674-685, 2009.

[12] Ito, T., Hiroyuki O., and Toramatsu S., "A Group Buy Protocol based on Coalition Formation for Agent-mediated E-Commerce". *IJCIS.* 3(1):11–20, 2002.

[13] He, L. and Ioerger T., "Combining bundle search with buyer coalition formation in Electronic Markets: A distributed approach through explicit negotiation", *Electronic Commerce Research and Applications*, 4(4), pp. 329-344, 2005.

[14] Chen, D., Jeng, B., Lee, W., and Chuang, C, "An agent-based model for consumer-to-business electronic commerce". *Expert Systems with Applications*, 34(1): pp. 469-481, 2008.

[15] Maniezzo, V., Gambardella, L.M. & Luigi, F. "Ant Colony Optimization". In: *Onwubolu, G.C. & Babu B.V. (eds): New Optimization Techniques in Engineering*, Vol. 14, 12004, pp. 101–117. Springer-Verlag, Berlin, Heidelberg.

[16] M. Dorigo and L.M. Gambardella, "Ant colony system: A cooperative learning approach to the traveling salesman problem," *IEEE Transactions on Evolutionary Computation,* vol. 1, no. 1, 1997, pp. 53–66.

[17] E. L. Lawler,J. K. Lenstra, A. H. G. Rinnooy-Kan, and D. B. Shmoys, *The traveling salesman problem*, New York, NY: Wiley, 1985.

[18] T. Yamada, and C.R. Reeves, "Solving the Csum permutation flowshop scheduling problem by genetic local search". in: *Proc. of 1998 IEEE International Conference on Evolutionary Computation*, pp. 230–234, 1998.

[19] D. V. Dang and N. R. Jennings, "Generating coalition structures with finite bound from the optimal guarantees," *In AAMAS,* 2004, pp. 564–571.

[20] M. Dorigo, L.M. Gambardella, *Ant colonies for the traveling salesman problem*, BioSystems 43, pp. 73–81, 1997.

[21] A. Colorni, M. Dorigo, and V. Maniezzo, "Distributed optimization by ant colonies", in: *Proc. of ECAL'91, European Conference on Artificial Life*, Elsevier Publishing,Amsterdam, 1991.

[22] M. Dorigo, V. Maniezzo, and A. Colorni, *The ant system: an autocatalytic optimizing process*, Technical Report TR91-016, Politecnico di Milano, 1991.

[23] M. Dorigo, *Optimization, learning and natural algorithms, Ph.D. Thesis*, Politecnico di Milano, Milano, 1992.

[24] J. Raka, T. Milan, and S. Dana, "Ant Colony Optimization Applied to Minimum Weight Dominating Set Problem", in *Proc. of the 12th WSEAS International Conference on Automatic Control (ACMOS '10), Modeling & Simulation*, Catania, Italy, May 29-31, 2010, pp. 322-326.

[25] Lee Yean Hooi, Mohd Azam Osman, Rosnah Idrus, Tan Shiang-Yen, "Acceptance Level of Push Technology-based Online Shopping Widget among Malaysians: Application of Technology Acceptance Theory", *WSEAS Transactions on Business and Economics*, Volume 7, pp. 211-220, 2010.

[26] H.-J. Yu, X.-Y. Li, and Y.-X Wang, "Pricing Strategy of Ecological Industry Chain Based on Game Theory", *WSEAS transactions on systems*, vol. 7, pp. 71-80, 2008.

[27] S. Chattrakul, "Advances in Practical Optimal Coalition Structure Algorithm", *Doctor of Philosophy Thesis, School of Science and Software Engineering, Faculty of Informatics*, University of Wollongong Thesis Collection s, 2010.

[28] Ab Hamid, NR., "Consumers' Behaviour Towards Internet Technology and Internet Marketing Tools", *International Journal of Communications*, Volume.2, No.3, pp. 195-204, 2008.

[29] Anna Veronica Baterina and Carlos Oppus, "Image Edge Detection Using Ant Colony Optimization", *International Journal of Circuits, Systems and Signal Processing*, Issue 2, Volume 4, pp. 25-33, 2010.

Anon Sukstrienwong received the B.Sc. in applied mathematics from King Mongkut's Institute of Technology Ladkrabang, Thailand in 1992. He received his master's degree in engineering from the University of Colorado, Denver, U.S.A. in 1994. He is currently an assistant professor in the department of computer and technology at Bangkok University in Bangkok, Thailand. His current research interests include: evolutionary multiobjective optimization, evolutionary algorithms in general.