An ant colony algorithm for the Quadratic Set Covering Problem

Mobina Masaeli
Department of Industrial Engineering
Amirkabir University of Technology
Tehran, Iran

Nasrin Ramshe
Department of Industrial Engineering
Amirkabir University of Technology
Tehran, Iran

***Abstract*— This paper develops an ant colony algorithm to solve the quadratic set covering problem. QSC is a combinatorial optimization problem. This problem, which arises in many applications, is an extension of set covering problem. We tune one of the algorithm's parameters by statistical test and examine the quality of the algorithm with a set of instances. Computational results show that our algorithm is efficient for median size of problem with a low density.**

***Keywords*— Quadratic Set Covering. Ant colony algorithm. ANOVA test.**

I. Introduction

According to Schilling et al. (1993), set covering problem (SCP) is the main model in covering problems. That is a fundamental combinatorial problem in Operations Research.

The SCP is important in practice, as it has been used to model a large range of problems arising from scheduling, manufacturing, service planning, information retrieval, etc. (Lan et al, 2007) and it has numerous applications in such diverse fields as job assignment in manufacturing, selection of operators, simplification of Boolean expressions or service location .(Bautista et al, 2007)

In general, in set covering problem, given m rows as clients must be allocated to at least one of potential facilities in given n columns with a specified non-negative opening cost c_j.

It has been given m rows as clients must be assigned to opened facilities and n columns, each with a specified non-negative cost c_j. Usually, at first, covering matrix is constructed based on DC and distances between of clients and facilities. $a_{ij} = 1$ if and only if row i can be covered by column j. In actual, the objective is covering each client (at least once) with a subset of sites of minimum total cost. In what follows, the model is formulated as a binary integer program.

$$\min \sum_{j \in J} c_j x_j \quad (1)$$

$$s.t.$$

$$\sum_{j \in J} a_{ij} x_j \geq 1 \qquad i \in I \qquad (2)$$

$$x_j \in \{0,1\} \qquad j \in J \qquad (3)$$

The binary variables x_j are equal to 1 if and only if column j belongs to the optimal solution. $I = \{1,2,...,m\}$ is the set of rows and $J = \{1,2,...,n\}$ is the set of columns. Constraint (1) ensures that each client is covered by at least one facility.

It is well known that this problem is NP-hard in the literature (e.g. Garey & Johnson, 1979), and many algorithms have been developed to solve it. Balas et al. (1996) and Fisher et al. (1990) solved this problem by exact algorithms that are mostly based on brand-and-bound and branch-and-cut. Beasley and Chu (1996) presented genetic algorithm and a few years later, Brusco et al. (1999) developed a simulated annealing algorithm. In that years, Ceria, Nobili, SassanoCaprara, Fischetti, and Toth proposed some effective heuristics based on Lagrangian relaxation together with subgradient optimization. Lan and DePuy (2006) solved this problem with improving a greedy algorithm by incorporating randomness and memory and Caserta (2007) applied tabu search algorithm. (Dorigo, Birattari, & Stützle, 2006; Dorigo & Di Caro, 1999; Dorigo & Stützle, 2004) was solved SCP by ACO. Bautista and Pereira (2007) used grasp algorithm for solving a special SCP case and in STS.

In the classic SCP, there is not a relation between the new located facilities but (Bazaraa and Goode, 1975) extend this problem to quadratic case where the objective function is modified. The quadratic set covering (QSC) can be expressed as follows:

$$\begin{aligned} &\min \sum_{j \in J} c_j x_j + \sum_{\substack{k, j \in J \\ j<k}} c_{jk} x_j x_k \\ &s.t. \\ &\sum_{j \in J} a_{ij} x_j \geq 1 \qquad i \in I \\ &x_j \in \{0,1\} \qquad j \in J \end{aligned} \qquad (4)$$

Non-negative cost c_{jk} is associated with relation between located facilities. It's value can be added to objective, when both facilities j and k has been opened.

This problem arises in many applications, such as the facility layout problem, the location of access points in a wireless network, line planning in public transports and etc.

Bazaraa and Goode (1975) developed a cutting-plane algorithm to solve certain quadratic set-covering problem. In 2007, Escoffier and Hammer studied the polynomial approximation properties of QSC and showed that this problem is very hard to approximate in the general case, and even in classical subcases. In spite of some applications of SCP, it has not been paid attention so much .In addition, since the SC is NP-hard, the QSC is NP-hard too and applying metaheuristics is reasonable for solving it.

ACO is a one of the metaheuristic algorithms which has been successfully applied to a great variety of hard COPs, so far. Ren, et. al. (2010), presented applying ACO is feasible for SCP and it can produce competitive solutions in comparison with other metaheuristics. Therefore, we apply it for solving the QSCP. The remaining of the paper is organized as follows: at the second stage, we present our algorithm. At the last we compare the result of the exact solution driven by GAMS software and solutions produced by ACO algorithm.

II. An ant colony algorithm for the problem

Ant colony algorithm is a mathematical model of ants' behavior in finding the shortest path between nest and food. The search capability of ants, using no visual sign, is the most attractive aspect of their behavior. Passing through the paths, each ant leaves pheromone on its path. The amounts of pheromone on different paths make ants able to improve the paths between nest and food they pass through. Ants' capability in improving and shortening the paths inspired scientist to develop algorithms for solving optimization problems (MiarNaimi, et. al., 2009)

In this algorithm, the distribution of the ants between potential sites is proportionate to cost per the number of related facilities to the special one. The rule of selecting next facility is according to equation (5).

$$\rho_k(r,s)=\begin{cases}\arg\max\left\{\left[\tau(r,u)^{\alpha}\right]\left[\eta(r,s)^{\beta}\right]\right\} & \text{if } q\le q_0\\ \dfrac{\left[\tau(r,u)^{\alpha}\right]\left[\eta(r,s)^{\beta}\right]}{\sum\left[\tau(r,u)^{\alpha}\right]\left[\eta(r,s)^{\beta}\right]} & \text{if } s\in J_k(r)\end{cases}\qquad(5)$$

The heuristic function has been defined as follows. We considered the best ant in all iteration updates the pheromone of his path. Moreover, we allowed that the worst ant participates in pheromone updating with a small probability. Pseudo code of this algorithm is shown in figure. 1.

Alpha is evaporation rate that has much effect on efficiency of the algorithm so we use ANOVA test for tuning this parameter as follows. A high value for beta parameter maybe leads to early convergence of algorithm.

The Set $S=\{0.28,0.48,0.68\}$ includes three different values of alpha parameter. There is a single-factor experiment, so we did a one way analysis of variance on the following data. The order of experiment was completely randomized.

The ANOVA test is significant at the 1 percent significance level since statistical Table F distribution shows 0.01 as the one percent $F_{0.99}$ for 2 and 24 degrees of freedom. We can then reject the hypothesis and claim that there is a considerable difference in value of cost function among the three values of alpha.

From the results in table 3, one sees that 'b' is differs significantly from 'a' and c, but 'a' and 'c' do not differ significantly from each other. The means of these three values for alpha have been shown in table 4. So, alpha has been set to value of second treatment that is 0.48.

III. Computational results

The algorithm was programmed in MATLAB software. All the runs were carried out on an Intel(R) Core(TM) 2 Dou CPU at 2.20GHz with 2.00GB of RAM system. The test instances are described in tables 1. The correlation costs between facilities are randomly generated using a uniform distribution between 100 and 150. Covering matrix is randomly generated by a function, as well. Each of the instances in each set has been run in 10 times and the average cost has been presented. Tables are shown at the end of paper.

IV. Conclusion and Future research

The QSC problem can be categorized as NP-hard class. As a result, metaheuristic algorithms can be used to solve this problem. In this respect, we solve the QSC problem with ant colony algorithm. Computational results have shown that the proposed ACO algorithm is efficient for median size instances and near optimal solutions can be obtained in plausible time duration.

In the strong sense, ant colony algorithm can be efficiently developed for the large scales QSC problem. On the other hand, quadratic set covering problem has not been addressed that much in the literature. In this regard, we propose that the problem can be meaningfully considered with some more practical assumptions and solved by other popular meta-heuristics such as genetic algorithm and simulated annealing. What is more, applying the ACO algorithm to produce an incipient population for GA algorithm seems fruitful. It seems good to apply the ACO for producing initial population for GA.

REFERENCES

[1] Balas, E., & Carrera, M. C. (1996).A dynamic subgradient-based branch-and-bound procedure for set covering. Operations Research, 44(6), 875–890.

[2] Bautista, J., & Pereira, J. (2007) .A GRASP algorithm to solve the unicost set covering problem. Computers & Operations Research, 34, 3162 – 3173.

[3] Beasley, J. E., & Chu, P. C. (1996).A genetic algorithm for the set covering problem. European Journal of Operational Research, 94(2), 392–404.

[4] Brusco, M. J., Jacobs, L. W., & Thompson, G. M. (1999). A morphing procedure to supplement a simulated annealing heuristic for cost- and coverage-correlated set-covering problems. Annals of Operations Research, 86, 611–627.

[5] Bazaraa, M. S. & Goode, J. J. (1975). A cutting plane algorithm for the quadratic set covering problem. Operations Research, 23(1), 150–158.

[6] Caserta, M. (2007). Tabu search-based metaheuristic algorithm for large-scale set covering problems. In K. F. Doerner et al. (Eds.), Metaheuristics: Progress in complex systems optimization (pp. 43–63). New York: Springer.

[7] Ceria, S., Nobili, P., & Sassano, A. (1998). A Lagrangian-based heuristic for large-scaleset covering problems. Mathematical Programming, 81, 215–228.

[8] Dorigo, M., Birattari, M., & Stützle, T. (2006). Ant colony optimization: Artificial ants as a computational intelligence technique. IEEE Computational Intelligence Magazine, 1(4), 28–39.

[9] Dorigo, M. & Di Caro, G. (1999). The ant colony optimization meta-heuristic. In D.Corne, M. Dorigo, & F. Glover (Eds.), New ideas in optimization (pp. 11–32).Maidenhead, UK: McGraw-Hill.

[10] Dorigo, M., & Stützle, T. (2004). Ant colony optimization. Cambridge: MIT Press.

[11] Escoffier, B., &L. Hammer, P. (2007). Approximation of the Quadratic Set Covering problem. Discrete Optimization, 4, 378–386.

[12] Garey, M. R. & Johnson, D. S. (1979). Computers and intractability: A guide to thetheory of NP-completeness. San Francisco, CA: Freeman.

[13] Lan, G. & DePuy, G. W. (2006). On the effectiveness of incorporating randomness and memory into a multi-start metaheuristic with application to the set covering problem. Computer & Industrial Engineering, 51(3), 362–374.

[14] MiarNaimi, H. & Taherinejad, N. (2009). New robust and efficient ant colony algorithms: Using new interpretation of local updating process. Expert Systems with Applications, 36, 481–488.

[15] Schilling, D. A., Jayaraman, V. & Barkhi, R. (1993). A review of covering problem in facility location.Location Science, 1(1), 25–55.

[16] ZanjiraniFarahani, R., Asgari, N., Heidari, N., Hosseininia, M. & Goh, M. (2012). Covering problems in facility location: A review. Computers & Industrial Engineering, 62, 368–407.

BIOGRAPHY

Mobina Masaeli is a MSc holder in Industrial engineering from Amirkabir university. She has done her BSc in Industrial engineering at K.N. Toosi University of Technology. Her research interests are supply chain management, network design and operations research.

Nasrin Ramshe obtained bachelor and master's degrees in Industrial Engineering fromAmirkabir University of Technology, Tehran, Iran. She is currently a Ph.D. student. Her research interests are operations research and material handling systems.

TABLE 1. DATA FOR ANALYSIS OF VARIANCE

	Alpha					
	a		b		c	
Cost function	3386.99	3446.38	3281.319	3414.356	3449.987	3412.835
	3325.75	3332.01	3313.242	3359.299	3403.210	3397.818
	3353.59	3309.10	3255.863	3305.291	3407.543	3350.248
	3395.82	3383.22	3329.515	3302.813	3398.244	3399.961
	3379.13	3433.97	3162.741	3370.153	3400.508	3393.458

TABLE 2. ALPHA DATA ANOVA

Source	DF	Sum of squares	Mean square	F value
Model	2	44700.1429	22350.0714	9.09
Error	27	66385.6619	2458.7282	
Corrected total	29	111085.8048		

TABLE 3. THE STUDENT-NEWMAN-KEULS (SNK)

SNK Grouping	Mean	N	Alpha
A	3401.38	10	3
A	3374.60	10	1
B	3309.46	10	2

TABLE 4. MEANS OF DATA FOR SECOND AND THIRD GROUPS

Number of Means	2	3
Critical Range	45.50018	54.981927

TABLE 5. TEST PROBLEMS OF THIS PAPER

Set	Rows(m)	Columns	Density (%)	No. of instances	Problem type
TP1	30	200	2	5	random
TP2	50	500	2	5	random
TP3	50	500	20	5	random
TP4	200	1000	5	5	random

TABLE 6. RESULTS OF PROBLEM INSTANCE

Set	Inst.	Obj. function	Avr. cost (ACO)	Time
TP1	1	2510.8764	2517.9026	0.145663
	2	2651.5278	2661.4241	0.163714
	3	2447.7592	2452.5982	0.138566
TP2	1	3246.9562	3281.3189	0.881989
	2	3229.7429	3257.7432	0.887136
	3	3587.7352	3618.3866	0.949598
	4	3594.6398	3626.7696	0.999854
	5	3601.4678	3806.0891	0.984839
TP3	1	1147.1783	1563.0142	0.881770
	2	1123.8533	1487.9785	0.961829
	3	1134.9329	1541.1388	0.596745
TP4	1	5362.8430	7702.3175	6.828457
	2	5834.8726	7935.8628	6.789348

```
Input : all parameters of the main problem
Input : α = 0.48, iter. = 1, β = 100, τ = C.ln(COF), η = (COF / C)
for i ∈ It
   ant = [];
   for j ∈ J
      if rand > 0.8
         min(C)
      else
         max(COF/C)
      end
      % Set covering test matrix check whether coverage is full or not
         while  find(SCT == 0 )
            q = rand();   pro = rand();
            if pro > 0.48  %find an uncoverd customer
            else
            end   %find all possible facilities covering the uncoverd client
               % choose one facility supposed to service the client
           if q ≤ Q
               max((τ × eta) * η × β)
           else
               S = max((τ × eta) * η × β);
               POF = S/∑S; % probability of each condidate facility
              % choose  a facility based on generating random probabilty
            end
         end % set covring test is completed
         ant(j).cost  % calculate the cost function
   end
         % find best ant in this iteration
         % saving the path of best ant
         τ = (1-α)×(τ + numel(ord)^5) + α/(best cost / bestbest) − 0.48×RFC     %  update τ
end
disp(bestcost);
```

Figure 1. Pseudo code of proposed ant colony algorithm