Journal of Modern Mathematics and Statistics 1 (1-4): 30-34, 2007
ISSN: 1994-5388

Enhancing the Shortest Route Relaxation of the Set Covering Problem

Farhad Djannaty
Department of Mathematics, Kurdistan University, Iran

Abstract: Lp relaxation of Ip problems usually provides good bounds on the objective function value IP problems, including SCP problems. Although, these bounds are strong they are time consuming and are not suitable to be used in a tree search algorithm. Quick and relatively strong bounds for IP problems have their own attractions. In this study, a new way is proposed to impose side constraints to a graph theoretic relaxation of SCP. Shortest Route Relaxation (SRR) of the Set Covering Problem (SCP) is upgraded by the reallocation of the column costs for some columns of the A matrix. This method is called Residual Cost Algorithm (RCA). This algorithm is applied to a strong cost allocation strategy. It is shown that the lower bound of the SCP is increased for a number of standard test problems and computational results are presented.

Key words: Network relaxation, set covering, residual cost, shortest route rebxation

INTRODUCTION

Set problems comprising set covering, set partitioning and set packing have attracted attention for many years and have applications in airline crew scheduling, bus crew scheduling, plant location, circuit switching, information retrieval, assembly line balancing, political districting and truck delivery Darby-Dowman and Mita (1985).

Let M= { 1, 2,. ..,m } be the set of m integers and let S denote a set of n subsets of M. Thus

N = { 1, 2, ..., n }

S = { s_1, s_2, ..., s_n } where $s_j \subseteq M$, $j \in N$.

Let

$$\left\{\begin{array}{l} a_{ij} = 1 \text{ if } i \in sj \\ aij = 0 \text{ if } i / \notin sj \end{array}\right\} (i = 1, ..., m , j = 1, ..., n)$$

The Set Covering Problem (SCP) can be defined as follows:

$$\text{Minimize} \sum_{j=1}^{n} c_j x_j$$

$$\sum_{j=1}^{n} c_j x_j \geq 1, i = 1, \ldots, m$$

$$x_j \in \{0, 1\}, j = 1, ..., n.$$

The decision variable x_j indicates whether s_j is selected or not and c_j is the cost associated with selecting s_j . The problem can be interpreted as finding the minimum cost selection of subsets of S such that each member of M is covered by at least one member of the selected subset of S.

If we replace the " ≥ " by " = " in each of the constraints of the above model, the modified problem is called the Set Partitioning Problem (SPP). If " ≥ " is replaced by " ≥ " and the objective function is to be maximized, the resulting model is the Set Packing Problem (SPK).

During the past 30-35 years a number of procedures have been developed which can deal with set problems. They can be found in Lemke *et al.* (1969), Salkin and Komcal (1973), Salkin (1975), Gerbracht (1978) and so on. They used either cutting plane algorithm and/or branch algorithm and then found that these algorithms are showed exponential and data dependent computing time Nemhauser and Wolsey (1988). Beasley (1987) has developed a tree search method to solve the SCP. Christofied (1975) reported good computational results with a steady state re-laxation method. Fisher and Kadia (1990) have developed a fast algorithm for a mixed set cov-ering/ partitioning problem. Of recent interest is the work of Harche and Thompson (1994) who have developed a new exact method called column subtraction.

Among the heuristic methods, Beasley (1990) has developed a Lagrangian heuristic which is reported to produce good quality results. Afif *et al.* (1995) have developed a new heuristic based on the ow algorithm of Ford and Fulkerson. Balas and Carrera (1996a) provide heuristics based on using Lagrangian relaxation embedded within branch-and-bound to solve the set covering problem. New heuristic approaches such as simulated annealing and neural networks have been tried Aourid and Kaminska (1994). Beasley and Chu (1996b) have done several modification to genetic procedures which produces high quality solutions. In recent years, there has been some advancements in solving NP-complete problems Xie and Wing (1998).

Set problems are categorized as NP-complete, which means that no polynomial time algorithm is known that guarantees to solve every instance of these problems. This increases the importance of relaxations which yield sharp lower and upper bounds.

SHORTEST ROUTE RELAXATION OF SCP AND SPP

Let k_j denotes the number of segments of ones in column a_j . A network (E,V) can be con-structed such that $V = \{ v_1, ..., v_m, v_m + 1\}$ where v_i corresponds to row i provided that a row m + 1 is defined to be identical to row m. For each column a_j associate a set of k_j arcs

$$E_j = \{ (v_{i1}, v_{i1}+l_1), ..., (v_{ik_j}, v_{ik_j} +l_{kj}) \} \; j = 1, ..., n$$

such that arc $(v_{ir} , v_{ir}+l_r)$ corresponds to rth segment in column a_j running from row i_r up to row $i_r + l_r - 1$. We can define E such that

$$E = \bigcup_{j=1}^{j=n} Ej$$

The two sets E and V as described above specify the structure of the shortest route relaxation of the Set Partitioning Problem (SPP). Let the cost of an arc from node v_p to node v_q be d^j_{pq}. A valid relaxation is obtained provided that

$$\sum_{(vp,vq)\in Ej} d^j_{pq} = c_j; j = 1, ..., n.$$

It can be verified that the Shortest Route Relaxation of set problems (SRR) is a proper relaxation.

For each arc (v_p, v_q) associated with a segment in column j , a binary variable $y^j_{pq} \in \{0, 1\}$ is defined such that y^j_{pq} equals 1 if this arc is included in the optimal route from the source node v_1 to the sink node v_{m+1} and equals 0 otherwise. Each feasible solution to the network problem is a route from v_1 to v_{m+1} and each feasible solution to the SPP problem corresponds to a route in the network problem such that if any arc corresponding to a segment in column j is selected , then all other arcs associated with column j must be selected as well. Thus SPP may be considered as a shortest route problem with side constraints and therefore SRR is a valid relaxation Associated with any feasible solution to the SPP problem there exists a route in the network from source to the sink, designated by

$$y^{*j}_{pq} .$$

Let x* be a feasible solution of the SPP , the corresponding route in the SRR network is constructed as follows:

$$\left.\begin{cases} \text{for } x^*j = 0 \quad \text{set} \quad y^{*j}_{pq} = 0, \forall(v_p, v_q) \in E_j \\ \text{and} \\ \text{for } x^*j = 1 \quad \text{set} \quad y^{*j}_{pq} = 1, \forall(v_p, v_q) \in E_j \end{cases}\right\} j = 1, ..., n$$

The shortest route relaxation for the Set Covering Problem (SCP) can be obtained by a similar procedure. To provide for possible overcovers let E, the arc set , be increased by adding m backward arcs $\{(v_i+1, v_i) \mid i = 1, 2, ..., m \}$ with costs

$$d^{n+i}_{(i+1)(i)} = 0$$

for i = 1, 2, ..., m . The introduction of these arcs creates cycles which allow rows to be overcovered.

THE RESIDUAL COST ALGORITHM (RCA)

The network associated with a given SPP or SCP is likely to have many multiple arcs since the same segment may appear in several different columns. Standard algorithms for finding the shortest route will necessarily only consider the arc of least cost from each set of multiple arcs. The set models considered to date do not contain additional constraints often known as side constraints. Such side constraints are not considered in the work presented here. The side constraints here may refer to the requirement that if any arc corresponding to a given column is included in the reduced network then all arcs from that column must also be included.

A reduced network is created by selecting a representative arc among all multiple arcs from node i to node j. The criterion for this selection is choosing an arc with the minimum arc cost, provided that the initial cost allocation is already completed. A tie in choosing the least cost arc is broken by selecting the first least cost arc encountered.

In RCA, the idea is to bring all arcs associated with a column j into the reduced network or, if this is not possible to drive all arcs of this column out of the reduced network. This is a new way to impose the side constraints to the shortest route relaxation. The Shortest Route Relaxation algorithm (SRR) should be applied on the reduced network to find the initial optimal route. Using notations proposed in section 2 each iteration of the RCA can be summarized as follows:

Let RNET be the set of all arcs comprising the current reduced network and let COLOPT be the set of all columns j having at least one arc in the current shortest route which is, obviously, a subset of RNET. A new cost allocation for all columns in COLOPT is obtained by adjusting arc costs according to rules determined by the current reduced network and the current shortest route. Let $dnet_{pq}$ be the reduced network cost or the least cost arc from node v_p to node v_q and $dcol^j_{pq}$ be the cost allocated to arc (v_p, v_q) associated with column j of the SCP, according to a predetermined cost allocation strategy. We will find the nonnegative residual cost $dcol^j_{pq}$-$dnet_{pq}$ for all k_j arcs obtained from the decomposition of column j. Let $SUM_j = \Sigma\ (v_p,v_q) \in E_j\ (dcol^j_{pq}-dnet_{pq})$. Based on the above residual costs an updated arc cost ($dcol'^j_{pq}$) should be calculated for each (v_p, v_q) associated with all columns j $\in$ COLOPT as follows.

$$dcol'^j_{pq} = \left\{ dnet_{pq} + \frac{SUM_j}{k_j} \right\}$$

Where k_j is the number of segments of ones in column j. It can be shown that this new cost alloca-tion strategy is also valid. Upon completion of this cost reallocation for columns of COLOPT all arcs in the previous shortest route will be equally above the current reduced network arc costs. In other words, the residual costs are distributed, equally, among all arcs. At this point the shortest route algorithm will be applied to the new reduced network which is likelv to have moqe costs on it's arcs and thus increases the lower bound of the SCP.

The RCA can be outlined as follows:

Apply the initial cost allocation on the full network
Construct the reduced network
Apply the SRR algorithm to find the inctial optimal route
repeat
    Find COLOPT, the columns contributing to SRR
        repeat
        Select a column j in COLOPT
        Fund all residual costs associated dith columnj
        Find SUMj
        Find all dcol0 j pq associated with column j
    Until all columns in COLOPT are processed.
    Update the reduced network.
    Apply the SRR algorithm on the redused network
    Until the iteration limit is reached.

A stopping criterion for RCA can be described as when the amount of reduction in the sum of the residual costs of all columns participating in the shortest route is smaller than a user specified constant. It is decided to stop at iteration i such that (RC(i) - RC(i - 1)) < k where RC(i) is the sum of the residual costs of all columns contributing to the shortest route in iteration I. There is considerable choice for the value of k. In order to take account of the problem specific characteristics, the value of k can be met equal to (RC(2) - RC(1))/170.

A numerical example: Let column j with cost cj = 65 be decomposed into 3 arcs (3,5), (12,13) and (20,22) with arc costs $dcol_{3,5}$ = 20, $dcol_{12,13}$ = 60 and $dcol_{27,72}$ = 15, respectively, where $dcol_{pq}$ refers to the arc cost associated with arc (v_p, v_q). If the leajt cost from node 3 to node 5 is $dnet_{3,5}$ = 16 and the least cost from node 12 to node 03 is $dnet_{12,13}$ = 30 and the least cost from node 20 to 22 is $dnet_{20,25}$ = 15, , we define the difference $dcol_{3,5}$ - $dnet_{3,5}$ = 4 as the residual cost of arc (5,5) of column j. Therefore, the corresponding residual costs of arcs (12,13) and (20,22) are 10 and 6, respectively. Now if we decide to distribute the residual costs equally among the 3 arcs, each arc will receive 16/3 and the new arc costs are 16+14/3,20+14/3 and 15+18/3. The new costs sum to 65 and the new cost allocation is therefore still feasible. If there is no other arc multiple to arc (20,22) with a cost between 15 and 15 + 14/3 then the new least cost in this arc increases from 15 to 15 + 14/3. Now if the arc (3 , 5) comes into the optimal route and the reduced network in the next iteration, the arc (12 , 13) comes in the reduced network but not in the optimal route and arc (20 , 22) is not in the reduced network and thus not in the optimal route. Therefore, the lower bound of RCA is increased at least by 14/3 and the new residual costs are 0, 0 and 14/3. The new arc costs after another iteration of RCP are 16+14/3+14/9 = 200/9, 20+14/3+14/9 = 236/9 and 15+14/9 = 149/9. As a result the reduced network costs are increased and the residual costs are decreased. If the same situation happens to this column in the next iteration the first two arc costs are increased by 14/27 and the third one is decreased by 28/27. RCA can be applied to columns associated with all arcs of the reduced network with a view to increasing arc costs of the network. On completion of this the topology of the reduced network is not changed but the arc costs are either the same or are increased. The same procedure can be applied to the new reduced network and a further increase in the arc costs can be achieved.

Computational results: A number of randomly generated SCP problems were collected from different sources. The charac-teristics of these problems are presented in Table 1. Problem AIR1 belongs to a set of 6 problems generated by Powers (1987) and used by El-Darzi (1988). Problems RDM3, RDM4, RDM6 and RDM7 are taken from

Table 1: Problem characteristics, LP and RCA execution times in PIII 800 seconds

Prob. Name	No. of rows	No. of columns	Ex. time of optimizer	Ex. time of RCA
AIR1	159	416	0.99	0.219
RDM3	101	109	0.88	0.109
RDM4	100	106	0.82	0.109
RDM6	100	130	0.93	0.109
RDM7	98	98	0.82	0.109
SCP51	200	2000	4.67	0.494
SCPA1	300	3000	35.77	1.868
SCPB1	300	3000	155.99	3.79
SCPE1	50	500	0.99	0.109
DUTY1	200	1000	1.76	0.164
DUTY2	200	1000	1.65	0.219
DUTY3	200	2000	2.36	0.274
DUTY4	200	2000	2.09	0.164
DUTY5	300	2000	3.52	0.494
DUTY6	300	2000	3.41	0.549

Table 2: Lower bounds obtained by LP and RCA

Prob. name	LU of St8 Bef. RCA	LU of St8 Aft. RCA	LP value	IP value
AIR1	12830	14829.4	16600	16610
RDM3	66.04	78.82	95.06	96
RDM4	61.11	77.68	93.57	97
RDM6	64.61	75.53	98.06	99
RDM7	56.4	68.64	86	87
SCP51	129.05	195.79	251.23	253
SCPA1	105.2	172.95	246.84	253
SCPB1	22.97	35.79	64.54	69
SCPE1	2.95	3.026	3.48	5
DUTY1	187.32	208.9	245	245
DUTY2	187.32	208.9	245.5	246
DUTY3	201.11	227.86	259	260
DUTY4	233.44	264.38	310	311
DUTY5	388.28	429.8	521.5	523
DUTY6	398.51	424.5	507	508

a set of 14 randomly generated problems supplied by Paixao (1983). Problem SCP51 is taken from a set of 25 problems from Balas and Ho and Problems SCPA 1, SCPB1 and SCPE1 are taken from a set of OR test problems provided by Beasley (1987). DUTY problems are two part duty crew scheduling problems generated by the author.

At this point the RCA algorithm is applied to the above mentioned test problems to improve the lower bound obtained by strategy 8 which is the strongest cost allocation strategy among 8 different cost allocation strategies proposed in Djannaty (1993) for the shortest route relaxation of the SCP. Computational results are summarized in Table 1 and 2. The number of rows and the number of columns are shown in columns 2 and 3 of Table 1. Columns 4 and 5 of this table show the time to reach the LP solution by FORTMP and the time to apply the cost all ocation strategy 8 and enhance it by RCA. Table 2 shows the lower bound obtained by strategy 8 before and after applying RCA and the LP and IP solutions of the test problems. In RCA the lower bound is nondecreasing and the stronger the initial lower bound, the more improvement in the final lower bound will be achieved.

As can be seen from Table 1 and 2, although the quality of the lower bound obtained by the LP relaxation is better than those obtained by RCA, the latter is much faster than the former. As the RCA is quick, the corresponding lower bounds can be incorporated in a tree search algorithm.

CONCLUSION

The research presented in this study, suggests a new way to impose side constraints on the shortest route relaxation of the set covering problem. The number of broken constraints are decreased in each iteration and thus better lower bounds can be obtained for the SCP problem. The RCA algorithm is much faster than LP relaxation of the problem and thus can be incorporated in a B and B algorithm. More investigations into finding stronger cost allocation strategies for the SRR are encouraged, in that, it can lead to stronger lower bounds by RCA and thus a more efficient tree searches.

REFERENCES

Afif, M., M. Hifi, V. Paschos and V. Zissimopoulos, 1995. A new efficient heuristic for the minimumset covering problem, J. Operat. Res. Soc., 46: 1260-1269.

Aourid, M. and B. Kaminska, 1994. Neural networks for the set covering problem: An application to the test vector compaction, IEEE Int. Conf. Neural Networks Conf. Proc., 7: 4645-4649.

Balas, E. and M.C. Carrera, 1996 A dynamic subgrsient-based Branch-and-Bound procedure for the set covering problem, Operat. Res., 44: 875-890.

Beasley, J.E., 1987. An algorithm for set covering problems, Eu. J. Operat. Res., 31: 85-93.

Beasley, J.E. and P.C. Chu, 1996. A genetic algorithm for the set covering problem, Eur. J. Operat. Res., 94: 392-404.

Beasley, J.E., 1990. A Lagrangian heuristic for set covering problems, Naval Research Logistics Quarterly, 37: 151-164.

Christofides, N., 1975. Graph theory: An Algorithmic Approach, Academic Press Inc.

Darby-Dowman, K. and G. Mitra, 1985. An extension of set partitioning with application to scheduling problems, Eur. J. Operat. Res., 21: 200-205.

El Darzi, E., 1988. Methods for solving the set covering and set partitioning problems using graph theoritic (relaxation) algorithms, PhD. Thesis, Brunel University.

Fisher, M.L. and P. Kedia, 1990, Optimal solution of set covering/partitioning problems using dual heuristics, Manag. Sci., 36: 674-687.

Gerbracht, R., 1978. A new algorithm for very large crew pairing problems, 18th AGIFORS Symposium, Vancouver, Canada.

Harche, F. and J.L. Thompson, 1994. The column subtraction algorithm: An exact method for solving weighted set covering, packing and partitioning problems, Comput. Operat. Res., 21: 689-705.

Lemke, C.E., H.M. Salkin and K. Spielberg, 1969. Set covering by single branch enumeration with linear programming subproblems, Operat. Res., 19: 998-1022.

Nemhausewr, G.L. and L.A. Wolsey, 1988. Integer and Combinatorial Optimization, Wiley.

Paixao, J., 1983. Algorithms for large scale set covering problems, Imperial College, London, internal report.

Powers, D., 1987. Investigation and construction of set covering and set partitioning test problems, BSc dissertation, Brunel University.

Salkin, H.M. and R.D. Koncal, 1973. Set covering by an all-integer algorithm: computational experience, J. Assoc. Comput. Machinery, 20: 189-193.

Salkin, H.M., 1975. Integer programming, Addison-Wesely.

Xie, J. and W. Xing, 1998. Incorporating Domain Specific Knowledge into Evolutionary Algorithms, Bejing Mathematics, 4: 131-139.