Computational comparison of two Lagrangian relaxation for the K-median problem

Ahn Sang Hyung

≪Contents≫

1. Introduction

The k-median problem has been widely studied both from the theoretical point of view and for its application. An interesting theoretical development was the successful probabilistic analysis of several heuristics (e.g. Fisher and Hochbaum [6] and Papadimitriou [11]) and relaxation (e.g. Ahn et at [2]) and polyhedral study (Ahn [1] and Guignard[8]) for this problem. On the other hand, the literature on the k-median problem abounds in exact algorithms. Most (e.g. Cornuejols et al [3]) are based on the solution of relaxtion.

The computational experience reported in the literature seems to indicate that this relaxation yields impressively tight bounds compared to what can usually be expected in integer programming. In this paper we perform computational analysis of two Lagrangian relaxation for the k-median problem.

Consider a set $Y=\{Y_1, Y_2, \cdots, Y_n\}$ of n points, a positive integer $k \leq n$ and let $c_{ij} \geq 0$ be the distance between Y_i and Y_j for each $1 \leq i \leq n$ and $1 \leq j \leq n$. The k-median problem consists of finding a set $S \subseteq Y$, $|S|=k$, that minimizes $\sum_{i=1}^{n} \operatorname{Min}_{j \in S} c_{ij}$ (Here $|S|$ denotes the cardinality of the set S.)

The k-median problem has the following integer programming formulation.

$$Z_{IP}=\operatorname{Min} \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \quad \cdots\cdots (1)$$

Author: Associate Professor, School of Management, Seoul National University

$$\text{s.t.} \sum_{j=1}^{n} x_{ij}=1 \text{ for } \quad i=1, 2, \ldots, n \cdots\cdots (2)$$

$$\sum_{j=1}^{n} y_j = k \cdots\cdots (3)$$

$$x_{ij} \leq y_j \quad \text{for } i, j=1, 2, \ldots, n \cdots\cdots (4)$$

$$x_{ij} \geq 0 \quad \text{for } i, j=1, 2, \ldots, n \cdots\cdots (5)$$

$$y_j \in \{0, 1\} \quad \text{for } j=1, 2, \ldots, n \cdots\cdots (6)$$

In this formulation $Y_j=1$ if $j \in S$, 0 othewise and, for $1 \leq i \leq n$, we can set $x_{ij}=1$ for an index that achieves $\underset{j \in S}{\text{Min}}\ c_{ij}$. Most successful exact algorithms reported in the literature are based on Lagrangian relaxation obtained by dualizing either constraint (3) or constraint set (4). In this paper we perform and compare computational experience of two Lagrangian relaxation on 3, 900 randomly generated test problems.

2. Lagrangian Relaxation

By dualizing assignment constraint set (2) with Lagrangian multipliers $u=\{u_1, \ldots, u_n\}$, we obtain Lagrangian relaxation.

(LR1)

$$Z_D(U) = \text{Min} \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}x_{ij} + \sum_{i=1}^{n} u_i(\sum_{j=1}^{n} x_{ij}-1)$$

$$= \text{Min} \sum_{i=1}^{n}\sum_{j=1}^{n} (c_{ij}+u_i)x_{ij} - \sum_{i=1}^{n} u_i$$

$$\text{s.t.} \sum_{j=1}^{n} y_j = k \cdots\cdots (3)$$

$$x_{ij} \leq y_j \quad \text{for } i, j=1, 2, \ldots, \text{n} \cdots\cdots (4)$$

$$x_{ij} \geq 0 \quad \text{for } i, j=1, 2, \ldots, n \cdots\cdots (5)$$

$$y_j \in \{0, 1\} \quad \text{for } j=1, 2, \ldots, n \cdots\cdots (6)$$

For fixed u_i's, above problem has the 0-1 VUB(variable upper bound) ructure. In order to solve (LR1), observe first that the objective function the (LR1) and the VUB constraints (4). These two imply that, for each i,

$$x_{ij}=\begin{cases}1, & \text{if } c_{ij}+u_i\leq 0\\ 0, & \text{otherwise.}\end{cases}$$

Hence with defining $\hat{c}_j=\sum_{i=1}^{n}\text{Min}(0,\ c_{ij}+u_i)$

(LR1) is equivalent to

$$\text{Min} \sum_{j=1}^{n} \hat{c}_j Y_j$$

$$\text{s.t.} \sum_{j=1}^{n} Y_j=k$$

$$Y_j\in\{0, 1\} \text{ for } j=1, 2, \ldots, n$$

which is a trivial problem. That is, optimal Y 's are

$$Y_j=\begin{cases}1, & \text{for the first } k \text{ smallest } \hat{c}_j\\ 0, & \text{otherwise.}\end{cases}$$

Since the objective is to minimize, clearly the best choice for u would be an optimal solution to the dual problem:

(D1)

$$Z_{D1}=\underset{U}{\text{Max}}\ Z_D(u)$$

By dualizing k-median constraint (3) with a lagrangian multiplier v, we have second lagrangian relaxation.

(LR2)

$$Z_D(v)=\text{Min} \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}x_{ij}+v(\sum_{j=1}^{n} y_j-k)$$

$$=\text{Min} \sum_{i=1}^{n}\sum_{j=1}^{n}c_{ij}x_{ij}+\sum_{j=1}^{n} vy_j-\sum vk$$

$$\text{s.t.} \sum_{j=1}^{n} y_j=1 \text{ for } i=1, 2, \ldots, n \cdots\cdots\cdots\cdots(2)$$

$$x_{ij}\leq y_j \quad \text{for } i, j=1, 2, \ldots, n \cdots\cdots\cdots(4)$$

$$x_{ij}\geq 0 \quad \text{for } i, j=1, 2, \ldots, n \cdots\cdots\cdots(5)$$

$$y_j\in\{0, 1\} \quad \text{for } j=1, 2, \ldots, n \cdots\cdots\cdots(6)$$

For fixed v, above problem is so-called SPLP(simple plant location problem). As is known, SPLP is not an easy problem to solve but admit highly efficient dual based algorithm (Krarup and Pruzan [10] and Erlenkotter[4]). So w adopt Erlenkotter's DUALOC to solve SPLP for given v.

Apparently, the best choice for v would be an optimal solution to the Lagrangian dual problem:

(D2)

$$Z_{D2} = \underset{V}{\text{Max}}\ Z(v)$$

Since (D1) and (D2) are subdifferentiable, we used subgradient method to solve these Lagrangian duals as proposed by Fisher [5]. Note that $Z_{IP} \geq Z_{D1}$, $Z_{IP} \geq Z_{D2}$ and we say there exists duality gap when $Z_{IP} \neq$ dual value. Because Z_D (u) is not increased by removing the integrality restriction on Y_j from the constraints of (LR1). $Z_{D1} = Z_{LP}$ (where Z_{LP} is the objective value of linear program relaxation of k-median problem). Geoffrion [7] calls this the integrality property.

(LR2) does not have the integrality property, so $Z_{D2} \geq Z_{LP}$. Thus $Z_{D1} \leq Z_{D2}$. That is, the lower bound obtained by (LR2) is tighter than that of by (LR1).

Two properties are crucial in evaluating a relxation.

(1) the tightness of the bound generated

(2) the amount of computational efforts required to get these bounds.

Usually there is a tradeoff between these two properties in choosing a relaxation. Tighter bound usually requires more computational efforts to get it than loose bound. However it is generally difficult to determine whether a relaxation with tighter bounds but great computational effort will end with better overall computational performance. That is, whenever there exists duality gap we have to resort to branch and bound technique to get an optimal solution. A branch and bound scheme incorporated with tighter bound requires smaller search tree than one with loose bound, i.e., if we spend more computational efforts to get an tighter bound, we could cut off the search tree fast. This is why extensive computational experience is needed to determine which relaxation is better in terms of overall computational performance.

3. Computational Experience

In this section, we report our computational experience with medium-size k-median problem. This computational experience is based on the solutions of 3,400 random problems with $n=50$ points and additional 500 random problems with $n=100$ points. As mentioned earlier, Z_{D1} and Z_{D2} were obtained by solving Lagrangian dual by subgradient optimization. If it happens that the value of the best known feasible solution equals the value of Lagrangian dual or all the subgradients equal 0, subgradient iteration terminates because we found optimum. For most of test problems with no duality gap, the algorithm terminated in less than 100 subgradient iterations because of the stopping criterion. If after 100 subgradient iterations, there was still a gap between the best feasible solution (an upper bound on Z_{IP}) and the best Lagrangian relaxation (a lower bound on Z_{IP}), we resorted to branch and bound to find Z_{IP}.

The first set of experiment involves unit edge length case with $n=50$ points. We generated 1,700 random graphs on which the k-median problem is defined. c_{ij} is the minimum number of edges on a path joining Y_i to Y_j for $1\leq i$, $j\leq n$, where the minium is taken over all paths joining Y_i to Y_j. Thus c_{ij} is the shortest distance between Y_i and Y_j, assuming that all edges have length one. In this case, when there exists a dominating set, Ahn et al [2] proved $Z_{IP}=Z_{D1}$. Therefore we expected first type of relaxation will do better computational performance. The results are summarized at Table 1. At (LR1) about 26% problems have duality gap and at (LR2) about 20% problems have duality gap as indicated by $Z_{IP}\neq Z_D$ at Table 1.

As was expected, the number of instances with duality gap are fewer in (LR2) than in (LR1). In (LR2), the number of instances with no duality gap is 1,359, whereas in (LR1) the number of instances with no duality gap is 1,265. However, (LR1) is better in terms of overall computational performance. This

Table 1. Unit Edge Length Graph

(total 1,700 problems)

value of k	LR 1				LR 2			
	$Z_{IP}=Z_{D1}$		$Z_{IP}\neq Z_{D1}$		$Z_{IP}=Z_{D2}$		$Z_{IP}\neq Z_{D2}$	
	no. of problems	CPU time	no. of problems	CPU time	no. of problems	CPU time	no. of problems	CPU time
2	97	1.322	73	2.401	125	2.118	32	4.010
3	97	1.002	73	2.203	129	2.565	29	3.999
4	109	1.004	61	2.129	121	2.330	37	3.810
5	127	1.231	43	2.308	123	2.056	35	2,978
6	125	1.361	45	3.007	129	2.305	29	3.917
7	135	1.589	35	2.566	123	1.946	35	3.645
8	139	1.809	31	2.897	115	2.920	43	4.712
9	139	2.062	31	2.910	109	2.643	49	5.244
10	151	2.198	19	2.998	120	2.906	38	4.982
11	146	1.715	24	4.100	145	2.296	13	4.111
total	1,265	—	435	—	1,359	—	346	—

(CPU time on VAX 11-780)

could be explained by the fact that even though duality gap exists usually it is very small. And that search through on the search tree does not require much efforts when compared to the efforts of getting Lagrangian dual.

The second set of experiment involves tree case with $n=100$ points. c_{ij} is the number of edges on the unique path from Y_i to Y_j. As Kolen [9] proved, dual ascent procedure for SPLP defined on a tree always finds optimum without entering into branch and bound phase. With this property and $Z_{D1}\leq Z_{D2}$. We expected second type relaxation would have computational edge over first type relaxation. The results are summarized at table 2. At (LR1) about 15.6% problems have duality gap and at (LR2) about 7.4% problems have duality gap as indicated by $Z_{IP}\neq Z_D$ at table 2.

As table 2 indicates, (LR2) has fewer instances with duality gap and has better over all computational perfermance. This is explained as follows. When the underlying structure on which the k-median problem is defined is tree (LR2) is SPLP on tree. Therefore, DUALOC always finds optimum for SPLP without entering into branch and bound phase. Moreover Lagrangian multi-

Table 2. Tree

(total 500 problems)

value of k	LR 1				LR 2			
	$Z_{IP}=Z_{D1}$		$Z_{IP}\neq Z_{D1}$		$Z_{IP}=Z_{D2}$		$Z_{IP}\neq Z_{D2}$	
	no. of problems	CPU time	no. of problems	CPU time	no. of problems	CPU time	no. of problems	CPU time
2	47	0.819	3	2.001	47	0.925	3	1.578
3	49	1.085	1	1.479	49	0.883	1	1.210
4	44	1.586	6	2.010	46	1.099	4	1.785
5	44	1.890	7	2.121	49	0.954	1	1.326
6	42	2.113	9	2.731	48	1.063	2	1.546
7	40	2.057	10	2.809	44	1.078	6	1.979
8	42	2.110	9	2.467	45	1.398	5	1.876
9	42	2.646	9	3.118	49	1.149	1	1.689
10	41	2.668	12	3.893	44	1.360	6	2.764
11	41	2.597	12	3.994	42	1.650	8	3.009
total	422	—	78	—	463	—	37	—

(CPU time on VAX 11-780)

Table 3. Random Graph

(total 1,700 problems)

value of k	LR 1				LR 2			
	$Z_{IP}=Z_{D1}$		$Z_{IP}\neq Z_{D1}$		$Z_{IP}=Z_{D2}$		$Z_{IP}\neq Z_{D2}$	
	no. of problems	CPU time	no. of problems	CPU time	no. of problems	CPU time	no. of problems	CPU time
2	164	1.002	6	1.103	170	1.996	—	—
3	154	1.345	16	1.832	165	2.567	5	2.742
4	152	1.724	18	2.168	165	1.844	5	1.913
5	149	1.777	21	3.125	157	3.382	13	3.883
6	138	1.983	32	4.167	157	2.299	13	2.732
7	131	2.167	39	5.132	161	1.865	9	2.157
8	128	2.203	42	7.851	150	2.239	20	4.251
9	125	2.421	45	4.334	143	2.574	27	4.330
10	125	2.407	45	5.49	141	2.475	29	5.219
11	133	2.442	37	7.096	137	2.43	33	5.911
total	1,399	—	301	—	1,546	—	154	—

(CPU time on VAX 11-780)

plier is only one in (LR2) but the number of multipliers in (LR1) is n.

The third set of experiment involves random edge length case with $n=100$

points. The edge lengths were computed as follows. The points were assigned random integer coordinates in a square of size 10×10 and the length of an edge was the Euclidian distance between its two end points, rounded to the closed integer. c_{ij} was taken to be the length of the shortest path joining Y_i to Y_j.

The results are summarized at table 3. At (LR1) about 17.7% problems have duality gap and at (LR2) about 9.1% problems have duality gap.

As Table 3 indicates (LR1) is better in overall computational performance with $k\leq5$ but (LR2) is better with $k\geq6$. In this case we can not conclude which relaxation is better in terms of overall computational performance.

Reference

1. Ahn, S., Polyhdral Study of *k*-Median Problem, 經營論集 XXII 4 (1988), 178-188.
2. Ahn, S., Cooper, C., Cornuejols, G. and Frieze, A., Probabilistic Analysis of a Relaxation for the *k*-Median Problem, Math. of Opers. Res 13(1988), 1-31.
3. Cornuejols, G., Fisher, M.L. and Nemhauser, G.L., Location of Bank Accounts to Optimize Float: An Analytic Study of Exact and Approximate Algorithms, Manag. Sci 23(1977), 789-810.
4. Erlenkotter, D., A Dual-based Procedure for Uncapucitated Facility Location, Opers. Res. 26(1978), 992-1009.
5. Fisher, M.L., The Lagrangian Relaxation Method for Solving Integer Programming Problems. Manag. Sci. 27(1981), 1-18.
6. Fisher, M.L. and Hochbaum, D.S., Probabilistic Analysis of the Planar *k*-median Problem, Math. of Oper. Res. 5(1980), 27-34.
7. Geoffrion, A.M., Lagrangian Relaxation for Integer Programming, Math. Programming Study 2(1974), 82-114.
8. Guignard, M., Fractional Vertices, Cuts and Facets of the Simple Plant Location Problem, Math. Programming Study 12(1980), 150-162.
9. Kolen, A., Location Problems on Trees and in the Rectlinear Plane, Mathematisch Centrum, Amsterdam, Netherland (1982).
10. Krarup, J. and Pruzan, P.M., The Simple Plant Location Problem: Survey and

Synthesis. Europ, J of Opers. Res. (1983).

11. Papadimitriou, C.H., Worst-case and Probabilistic Analysis of a Geometric Location Problem, SIAM J. Comput. 10(1981), 542-557.