On Constraint Clustering to Minimize the Sum of Radii

Rajkumar Jain, Narendra S. Chaudhari

Abstract: ***We consider the min-cost k-cover problem: For a given a set P of n points in the plane, objective is to cover the n points by k disks, such that sum of the radii of the disks is minimized. In this paper we introduce the concept of constraints for min-cost k-cover problem. In any instance I of k-cover, the optimal solution value is at most the maximum radius r of ball $\boldsymbol{B(v,r)}$ centered at $v \in V$ in I. It implies that, in optimal solutions there always exists a constraint that separates the optimal solution. Investigation formulate that a can-not link constraint always separate the optimal solution very clearly and reduces cardinality of distinct maximal discs. Introduction of constraints improves the performance of min-cost k-cover algorithm over the existing algorithms.***

Keywords: ***k-clustering, min-cost k-cover, minimum sum of radii cover, constraint clustering.***

I. INTRODUCTION

Clustering analysis is fast emerging field of research in the vast arena of data mining, machine learning, operation research, and its allied areas. Cluster analysis [1-4] is mainly concerned with the problem of partitioning a given set of entities into homogeneous and well-separated subset, such that similar objects are kept in a group whereas dissimilar objects are in different groups. Constraints facilitate hands on information about the desired partition and strengthen performance of clustering algorithms. The primary function of constraint based algorithms is not to encompass all the domain expert's requirements but also instrumental in directing the algorithm to a desirable set partition by adopting user specified or derived constraints whereas constraint algorithm stimulates composition of a desirable clustering of the instances. In this paper we consider the min-cost *k*-cover problem. The Euclidean min-cost *k*-cover problem defined as follows. Given a metric *d* defined on a set *V* of points, we define the ball $B(v,r)$centered at $v \in V$ and having radius $r \geq 0$ to be the set $\{q \in V \mid d(v,q) \leq r\}$. In the minimum cost *k*-cover problem, we are given a set *P* of *n* points and integer *k* (*k* > 0). For *κ > 0*, computing a *κ*-cover for subset $Q \subseteq P$ is a set of at most *κ* balls covering all point of set *Q* and each ball centered at a point in *P*. The sum of the radii of balls is the cost of a set *D* of ball denoted by cost(*D*). In the Euclidean version, *P* is given as a set of points in some fixed dimensional Euclidean space R^l, and *d* is the standard Euclidean distance. In the metric version of the min-cost *k*-cover problem, we have *P* and *k* and the distance *d* between every pair of points in *P*.

Manuscript published on 30 August 2016.
***** Correspondence Author (s)

Rajkumar Jain, Department of Computer Science & Engineering, Indian Institute of Technology, Indore (M.P), India.
Narendra S. Chaudhari, Department of Computer Science & Engineering, Indian Institute of Technology, Indore (M.P), India.

The min-cost *k*-cover problem is well studied in general and Euclidean metrics. The initial results on approximation algorithms are due to Doddi *et al*. [5] who considered the metric min-cost *k*-cover problem and the closely related problem of partitioning *P* into a set of *k* clusters so as to minimize the sum of the cluster diameters they call this clustering to minimize the sum of diameter (MSD). They showed that unless $P = NP$ for any $\epsilon > 0$ there is no, $(2-\epsilon)$-approximation algorithm for the MSD problem. They also present a bicriteria polynomial time algorithm that returns *O*(*k*) clusters whose cost is within a multiplicative factor *O*(log(*n*/*k*)) of the optimal. For the objective function of minimizing the *k*-cover problem, Charikar and Panigrahy [6] presented a polynomial time algorithm based on the primal-dual method that gives a constant factor approximation algorithm of around 3.504 and thus also a constant factor approximation for MSD problem. Proietti and Widmayer [7] consider a problem closely related to the metric min cost *k*-cover problem. They prove that the problem is NP-hard, but for fixed *k* they give polynomial time algorithms. Lev-Tov and Peleg [8] research work is concerned with geometric disk covering problem. They give a polynomial time approximation scheme, having approximation factor of (1+6/*k*) and time complexity of $O(k^2.(nm)^{\gamma+2})$, for a problem that is closely related to the geometric min-cost *k*-cover problem and they call this minimum sum of radii cover (MSRC) problem. Bilo *et al.* [9] give polynomial time approximation schemes for generalizations of the MSRC problem. For minimizing the α^{th} power of the radii of the balls, where $\alpha \geq 1$ they give approximation schemes for a generalization of the MSRC and they also show that for $\alpha \geq 2$ such a generalization min-cost *k*-cover problem in the plane and the MSRC problem are NP-hard. They also give a polynomial time algorithm of $O(n^{O(\lambda^4+\xi)})$ if points lie on a line. Alt *et al.* [10] show that the NP-hardness result for the MSRC problem can be extended to any $\alpha > 1$. Gibson *et al.* [11] give an $(1+\epsilon)$-approximation algorithm for metric version of min-cost *k*-cover problem. Gibson *et al.* [12] give an exact algorithm which runs in time $O(n^{(logn.log\Delta)})$ for the metric version of min-cost *k*-cover problem. They also give $(1+\epsilon)$-approximation algorithms for metric version of min-cost *k*-cover problem when model of computation does not hold. Gibson *et al.* [13] designed an polynomial time exact algorithm for euclidean min-cost *k*-cover problem which runs in time $O(n^{881}.T(n))$, where $T(n) \geq 1$ is an upper bound on the time needed to compare the costs of two subsets of *D*. Behsaz *et al.* [14] give polynomial time exact algorithm for the unweighted minimum sum of radii problem when no singletons (clusters of radius zero) is allowed.

In the present research, we introduced the concept of constraints in min-cost k-cover problem. We modify the $O(n^{881}.T(n))$ polynomial time exact algorithm of Gibson *et al.* [13] to obtain a polynomial time constraint based algorithm for euclidean min-cost k-cover problem. The present research identify the can-not link constraints and apply these derived constraints in the algorithm for the reduction of distinct maximal disks and reduction of all enumerated subsets of the distinct maximal disks in minimum sum of radii problem. This in turn reduces the number of table entries of exact algorithm [13]. The basis for the constraint technique is motivated by a observation that in any instance I of k-cover, the optimal solution value is at most the maximum radius r of ball $B(v\ ,r)$ centered at $v \in V$ in I. It implies that, in optimal solutions there always exists a constraint that separates the optimal solution.

II. PRELIMINARIES

Research commence with the observation that for any instance I of k-cover, the solution value is at most the maximum radius r of ball $B(v\ ,r)$ and optimal solution is decidedly separable. Thus it permits to compute an optimal k-cover efficiently using constraints. We call a ball of zero radius as singleton ball. Similarly, we call a disc of zero radius as singleton disc.

Lemma 1. *For any instance of I of k-cover, the optimal solution value is at most the maximum radius r of ball B(v, r) in I.*

Proof. Given any instance of k-cover, a solution $B = \{B_1, B_2, \dots, B_k\}$ consisting of k balls covering all instance of I. The cost of B is given by the following formula $\text{cost}(B) = \sum_{i=1}^{k} radius(B_i)$. Assume balls B_2, B_3,...,B_k as singleton balls. Assign randomly chosen $n - k + 1$ points to ball B_1 and assign remaining $k - 1$ points to $k - 1$ balls such that each ball contains single point. Hence, $\text{cost}(B) = \text{cost}(B_1) = radius(B_1)$. Therefore the the optimal solution value is at most the maximum radius r of ball $B(v, r)$ in I.

Definition 1: A distinct maximal disc (DMD) is a disc if one cannot add any point to it without increasing its radius. Any solution can be reduced into one having only distinct maximal disc without increasing the cost. Thus, non-distinct maximal discs can be ignored to obtain optimal solution.

Lemma 2. In the min-cost k-cover problem, the number of distinct maximal discs is at most $O(n^2)$.

Proof. Let us consider a set of points $P = \{v_0, v_1, \dots v_l\}$ and $r_0, r_1, \dots r_n$ are the sorted distance from the point v_0 in the ascending order. For any value of $1 < i \leq l$, consider a ball $B(v_o, r)$ of radius r, such that $r_i \leq r < r_{i+1}$. It implies that distinct maximal discs $B(v_0, r_i) = B(v_0, r)$. So, the only distinct maximal disc centered at v are $B(v, r_i)$ for $1 \leq i \leq l$ and v is center of $l \leq n$. Each point can be the center of at most n distinct maximal discs, distinct maximal discs also include disk of radius zero or singleton disk, and therefore there are at most n^2 distinct maximal discs. The number of distinct maximal discs for the min-cost k-cover problem is at most n^2 (by Lemma 2).This yields a very important advantage when we deal with the min-cost k-cover problem. All subsets of the distinct maximal disc having size at most l can be enumerated in time $O(n^{2l})$. When l is a constant, this is polynomial time.

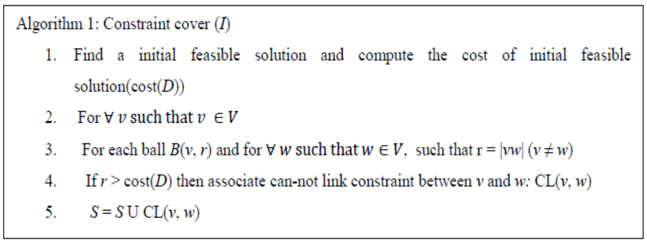
III. CONSTRAINTS BASED MIN-COST K-COVER APPROACH

Wagstaff and Cardie [15] introduced constraints in the area of data mining research. There are two types of constraints that were termed as must-link constraint and can-not link constraint. In must-link (ML) constraint two instances have to be in the same group, ML(a, b) symbolize instance a and b to have be in the same group. In cannot-link (CL) constraints two instances must not be placed in the same group, CL(a, b) symbolize instance a and b to have be in the different group. Algorithm 1 finds can-not link constraints in any instance I of k-cover problem.

3.1. Constraint Algorithm

Input: Instance of min-cost k-cover problem, A set S of can-not link constraint, $S = \varnothing$

Algorithm 1 takes input as any instance of I of min-cost k-cover problem. Computes initial feasible solution in accordance of Lemma 1. Can-not link constraints are investigated and generated for each ball $B(v, r)$, where $v \in V$ and r is the radius of ball. Algorithm 1 return a set of can-not link constraints.



```
Algorithm 1: Constraint cover ($I$)
    Find a initial feasible solution and compute the cost of initial feasible solution(cost($D$))
     For $\forall\ v$ such that $v \in V$
      For each ball $B(v, r)$ and for $\forall\ w$ such that $w \in V$, such that r = $|vw|$ ($v \neq w$)
       If $r >$ cost($D$) then associate can-not link constraint between $v$ and $w$: CL($v$, $w$)
        $S = S \cup$ CL($v$, $w$)
```

Theorem 1: *Maximum number of can-not link constraints generated in any instance I of k-cover problem is order of O(n^2).*

Proof. We have instance I of k-cover problem consisting of n points. For any two points p, q can-not link constraint exist in$B(v, r)\ if\ r > |pq| > \text{Icost}(D)$. Formally it can be stated as $\exists$ CL $-$ constraint, If $(|pq| > \text{Icost}(D))$, $\forall\ p, q : p, q \in V$. Point p and q can be any point from the V, so maximum n^2 combination can be possible. This is similar to the finding the distinct maximal discs. The number of distinct maximal discs is at most n^2 (by Lemma 2). Therefore, Maximum number of can-not link constraints is order of $O(n^2)$.

Theorem 2: *Minimum number of can-not link constraints generated in any instance I of k-cover problem is order of O(*k*).*

Proof. For k =1, all points of I are covered by a single ball $B(v, r)$. Assume that a can-not link constraint CL(p, q) exist between any two points p, q of ball $B(v, r)$, then p, q together $\exists\ CL(p, q)|\ p, q \in B(v,\ r)$then ($p$ and ($n - 2$)points) $\in B_1(v,\ r_1)$ and point $q \in B_2$ can-not Then, $n - 1$ points belong to $B_1(v, r_1)$ and point q belong to singleton disk B_2. In this manner if there are $k - 1$ can-not link constraints then k balls can cover n points, $n - k + 1$ points are covered by a ball B_i (v, r_i) and rest of the k-1 points are covered by k singleton disks. Solution obtained in this way is a feasible and optimal solution (by Lemma 1).

Theorem 3: *Number of discs in reduced distinct maximal discs is always less than distinct maximal discs, $|D_R| < |D|$.*

Proof. In the minimum sum of radii problem, the number of distinct maximal discs is at most $O(n^2)$ and also optimal solution contain discs only from the set of distinct maximal discs . Let D_r denotes the reduced distinct maximal discs. There are always exist at least a can-not link constraint in any instance of min-cost k-cover problem (by Theorem 2). If a can-not link constraint belongs to distinct maximal disc then that distinct maximal disc will not be part of optimal solution. So, with the help of can-not link constraints, it can checked be whether a distinct maximal disc will be a part of canonical optimal solution or not. Hence, can-not link constraints reduce the number of distinct maximal cluster. Let S denote set of can-not link constraints and α denoted the cardinality of set S such that, then maximum value of α is $O(n^2)$ (by Theorem 1). If α constraints ($O(n^2)$) are applied on the distinct maximal discs $D(O(n^2))$ then D will be reduced by considerable factor, hence $|D_R| < |D|$.

Corollary 1: *A can-not link constraint clearly separate the optimal solution.*

3.2. Constraint based min-cost k-cover algorithm

Consider an instance of the Euclidean min-cost k-cover problem which consists of a set of points V on the plane along with an integer k. Here, the distance between any pair of points is the Euclidean distance of these points in the plane. Let D be the set of distinct maximal discs with a center $p \in V$ and radius $|pq|$ for some $q \in P$. D includes the disk of radius 0, thus $|D| = n^2$.

Gibson et al. approach [13]: An axis parallel rectangle is called balanced if the ratio of its width to length is at least 1⁄3. This approach uses balanced rectangles to define the sub-problems. A separator for a (balanced) rectangle R is any line which is perpendicular to its longer side and cuts it in the middle third of its longer side of the rectangle R. The algorithm starts with a rectangle R_0 containing all the points and cuts it into two smaller rectangles by selecting a separator line and solves the sub-problems recursively. The A vertical or horizontal line is called critical if it either goes through a point $p \in$ P or if it is tangent to some disk in D. All vertical lines between two consecutive critical vertical lines intersect the same set of discs. Thus, there are only $\Theta(n^2)$ vertical or horizontal lines to consider as separators. To get an optimal solution it is required to consider only $|T| \leq \beta = 424$. It signify that the size of the dynamic programming table is $O(n^{2\beta+5})$, which is polynomially bounded.

Constraint algorithm: A set S of can-not link constraints are generated using Algorithm 1. The constraint min-cost k-cover algorithm takes input a rectangle R, an integer $\kappa \geq 0$, a subset $T \subseteq D$, a set of constraint S and has a recursive procedure $\mathrm{DC}(R, \kappa, T, S)$, computes an optimum solution using at most κ discs for the set of points in $Q = \{q: q \in P \cap R, \mathrm{q\ is\ not\ covered\ by}\ T\}$. A can-not link clearly separate the optimal solution (by Corollary 1). This implies that a separator is simulating like a can-not link constraint and diving the problem into sub-problem and solving it recursively. The algorithm calls $\mathrm{DC}(R_0, k, \emptyset, S)$ to find the best cover for P by applying α constraint's. The value of the sub-problem for a recursive call is stored in a dynamic programming table $\mathrm{Table}(P \cap R, \kappa, T)$. In constraint algorithm initial steps are basic initialization step and it remains same as constraint algorithm follows the steps as of algorithm [13]. Numbers of separators are directly proportional to the number of distinct maximal disk. Our based constraint approach uses the reduced distinct maximal disk instead of distinct maximal disk so number of separators will be reduced.

The overall running time of a call to $\mathrm{DC}(R_0, k, \emptyset, S)$ is bounded by table entries. Each table entry is indexed by a set of points $P \cap R$ for some balanced rectangle R, a $\kappa \leq k$ and a set $T \subseteq D$ such that $|T| \leq \beta = 424$. The proposed constraint based approach reduces the β hence, running time also reduces. Number of disk inside R, intersected by a separator is at most 12(Lemma 2.1 in [13]). Our constraint based algorithm uses the reduced distinct maximal disks (D_r) instead of distinct maximal disks (D) so β will be reduced which in turn improves the performance of algorithm.

IV. CONCLUSION

Constraint based approach to min-cost k-cover problem improves the bound of algorithm. The research portrays how constraint based algorithm is convenient and can yields better empirical results in comparison to non-constraint algorithms for min-cost k-cover problem. Number of calls stored in a table bounded by $O(n^{881})$ are very high for small value of n. In the present research, can-not link constraints reduced the size of the table entries. The research can be concluded that with minimizing the number of distinct maximal disks number of entries in the table are reduced, which in turn improves the overall complexity of the algorithm significantly.

REFERENCES

1. Cormack R M. A review of classification. Journal of the Royal Statistical Society, Series A. 134 (1971) 321–367.
2. Hartigan J A. Clustering Algorithms. John Wiley & Sons, New York, 1975.
3. Anderberg M R. Cluster Analysis for Applications. Academic, New York, 1973.
4. Hansen P, Jaumard B. Minimum sum of diameters clustering. Journal of Classification. 4 (1987) 215–226.
5. S. R. Doddi, M. V. Marathe, S. S. Ravi, D. S. Taylor, and P. Widmayer, Approximation algorithms for clustering to minimize the sum of diameters, Nordic Journal of Computing. 7 (2000) 185–203.
6. M. Charikar and R. Panigrahy, Clustering to minimize the sum of cluster diameters, Journal of Computer Systems Science. 68 (2004) 417–441.
7. G. Proietti and P. Widmayer, Partitioning the nodes of a graph to minimize the sum of subgraph radii, in Proceedings of the International Symposium on Algorithms and Computation (ISAAC), 2005, pp.578–587.
8. N. Lev-Tov and D. Peleg, Polynomial time approximation schemes for base station coverage with minimum total radii, Computer Networks. 47(2005) 489–501.
9. V. Bilo, I. Caragiannis, C. Kaklamanis, and P. Kanellopoulos, Geometric clustering to minimize the sum of cluster sizes, in Proceedings of the European Symposium on Algorithms, Lecture Notes in Computer Science 3669, Springer, New York. 3669 (2005) 460–471.

10. H. Alt, E. Arkin, H. Bronnimann, J. Erickson, S. Fekete, C. Knauer, J. Lenchner, J. Mitchell, and K. Whittlesey, Minimum-cost coverage of points by disks, in Proceedings of the Annual Symposium on Computational Geometry, 2006, pp. 449–458.
11. Gibson, M., Kanade, G., Krohn, E., Pirwani, I.A., Varadarajan, K.: On clustering to minimize the sum of radii. In: SODA, SIAM, Philadelphia,2008, pp. 819–825.
12. Gibson, G. Kanade, E. Krohn, I. Pirwani, and K. Varadarajan, On metric clustering to minimize the sum of radii, Algorithmica. 57 (2010) 484–498
13. Gibson, M., Kanade, G., Krohn, et al. On clustering to minimize the sum of radii. SIAM Journal on Computing, 41 (2012) 47-60.
14. Behsaz B, Salavatipour M. R. On Minimum Sum of Radii and Diameters Clustering. In: Fomin F V, Kaski P ed. Proceeding of 13th Scandinavian Symposium and Workshops, Helsinki, Finland, Algorithm Theory – SWAT 2012, Lecture Notes in Computer Science, Springer. 7357 (2012) 71-82.
15. Wagsta K, Cardie C, Clustering with Instance-level Constraints, Proceedings of the Seventeenth International Conference on Machine Learning, 2000, pp. 1103-1110.