Indian Statistical Institute

Kolkata

M.Tech. (Computer Science) Dissertation

Minimizing Maximum Movement to attain Connectivity

A dissertation submitted in partial fulfillment of the requirements
for the award of Master of Technology
in
Computer Science

Author:
Badrinath Sharma
Roll No:MTC1206

Supervisor:
Prof. Sandip Das
Advanced Computing and
Microelectronics Unit

M.TECH. (CS) DISSERTATION THESIS COMPLETION CERTIFICATE

Student : Badrinath Sharma (mtc1206)

Topic : Minimizing Maximum Movement to attain Connectivity

Supervisor : Sandip Das

This is to certify that the thesis titled Minimizing Maximum Movement to attain Connectivity submitted by Badrinath Sharma in partial fulfillment for the award of the degree of Master of Technology is a bonafide record of work carried out by him under our supervision. The thesis has fulfilled all the requirements as per the regulations of this Institute and, in our opinion, has reached the standard needed for submission. The results embodied in this thesis have not been submitted to any other university for the award of any degree or diploma.

Prof. Sandip Das

Date : 30^{th} *June*, 2014

Acknowledgements

At the end of my dissertation and my M.Tech. training at the Indian Statistical Institute Kolkata, I want to thank and give credit to all individuals who have provided me with invaluable assistance. Whether be it gentle guidance or access to materials or services that helped me a lot in my research work, it is greatly appreciated.

First and foremost I offer my sincerest gratitude to my supervisor, Prof. Sandip Das, who has supported me throughout my thesis with his patience and knowledge. It was a memorable learning experience. For his patience, for all his advice and encouragement and for the way he helped me to think about problems with a broader perspective, I will always be grateful. One simply could not wish for a better or friendlier supervisor.

I would like to thank all the professors at the Indian Statistical Institute Kolkata who have made my educational life exciting and helped me to gain a better outlook on computer science. I would like to express my gratitude to Ayanda, Soumenda, Gopi and Ratan for interesting discussions. Without the precious suggestions from Ayanda, it would have been considerably difficult to finish this report.

I would like to thank everybody at Indian Statistical Institute, Kolkata for providing a wonderful atmosphere for pursuing my studies. I thank all my classmates, seniors and juniors who have made the academic and non-academic experiences very delightful.

My most important acknowledgement goes to my family and friends who have filled my life with happiness.

Abstract

Given n discs of same radius in R^2, we study the problem of moving the discs to form a connected configuration such that the maximum moving distance of the discs is minimized. It appears to us that it is a difficult problem. Three discs and four discs examples were seen. We put the discs in a line and connect them with the same objective. The disc here could be any object or pebble. The pebbles are placed in the vertices of a tree and the movement is allowed only along the edges of a tree. Here also we need to move the pebbles to get them in a connected configuration. In general, this work investigates the movement problems; primarily focussing on minimizing the maximum movement. This will help determine the amount of battery life required by the devices to come to a connected state to impart some information.

Contents

List of Figures

Chapter 1

Introduction

The basic question that calls for the study on movement problems is : What is the minimum maximum movement required to be made by any device or object under consideration to attain some desired property? This desired property could be anything, it is connectivity in our case.
Consider an army of soldiers surrounding a terrorist camp to get them in custody. A reliable but short-range radio is given to each soldier. There is limited connectivity to a satellite (or other central location) for sharing the approximate positions of soldiers.
To form an effective communication network (for voice or data traffic), the radios retained by the soldiers must form a connected graph. This scenario subsequently leads to the problem of determining the minimum distance (time) each soldier needs to move to reach a configuration that induces a connected radio network. Any two soldiers can talk to each other in the reliable radio network, possibly using multiple hops. However, autonomous robots and not soldiers are considered the objects because of energy and resource constraints. The autonomous robots hold limited wireless connectivity and limited mobility in the field and the purpose is to minimize the use of these resources to form a reliable radio network.
This work is a study on a class of problems known as Movement problems, first introduced in [Demaine09] and further investigated in [Demaine009,Friggstad08]. In particular, we have focused on ConMax which includes minimization of maximum movement to reach connectivity. This basic connectivity problem has many variations. For example, ConSum involves minimization of the total movement while ConNum deals with minimizing the number of devices to be moved.
Consider n circular discs of radius r primarily scattered in a 2-D plane. Our goal is to get them in a connected configuration such that the maximum movement among the discs is minimized. Demaine et.al. [Demaine09] have given approximation algorithms for graphical setting where movement is restricted only to edges and objects that seek to be connected with each other are placed in some of the vertices. These objects can be denoted as pebbles. In the graphical setting, our objective is to minimize maximum movement to ensure that the subgraph of the vertices occupied by the pebbles is connected.
It is generally difficult to connect the discs of same radius in 2-D plane when minimizing the maximum movement. So we tried to solve the problems with some assumptions. We started with a linear problem i.e. all the discs of same radius were place in a collinear manner. Three discs, four discs problem in 2-D were seen. The discs placed in the circumference of a sufficiently large circle were studied. We placed discs on some vertices of tree, here the movement of discs is restricted only along the edges of the given tree.

Chapter 2

Related Works

Danny Z. Chen et.al.[Chen] have studied the problem of moving n sensors on a line to form a barrier coverage of a specified segment of the line such that the maximum moving distance of the sensors is minimized. It was an open question whether this problem on sensors with arbitrary sensing ranges was solvable in polynomial time previously. By giving an $O(n^2 logn)$ time algorithm they had settled the open question. For the special case when all the sensors have the same-size sensing range, the previously best solution takes $O(n^2)$ time. They have given an $O(nlogn)$ time algorithm for this case ; further, if all sensors are initially located on the coverage segment, their algorithm takes $O(n)$ time. Also they have extended their techniques to the cycle version of the problem where the barrier coverage is for a simple cycle and the sensors are allowed to move only along the cycle. For sensors with the same-size sensing range they have solved the cycle version in $O(n)$ time, improving the previously best $O(n^2)$ time solution.

Demaine et.al.[Demaine09] have given approximation algorithms and inapproximability results for a class of movement problems. In general, these problems involve planning the coordinated motion of a large collection of objects to achieve a global property of the network while minimizing the maximum or average movement. In particular, they consider the goals of achieving connectivity, achieving connectivity between a given pair of vertices, achieving independence and achieving a perfect matching. ConMax problem is one which minimizes maximum movement to reach connectivity, they have given $O(\sqrt{(m/OPT)})$ approximation algorithm in a graph or in the Euclidean plane, where m is number of pebbles and n is the number of vertices in the graph. They also have given evidence that even the geometric scenario is difficult. Given a tree T and a configuration of k pebbles on T, ConMax can be solved in polynomial time. We are giving an $O(n)$ time algorithm for this.

Demaine et.al.[Berman] again develop a constant-factor approximation algorithms for minimizing the maximum movement made by pebbles on a graph to reach a configuration in which the pebbles form a connected subgraph (connectivity), or interconnect a constant number of stationary nodes (Steiner tree).
Czyzowicz et.al.[Czyzowicz] have considered n mobile sensors that are located on a line containing a barrier represented by a finite line segment. Sensors form a wireless sensor network and are able to move within the line. An intruder traversing the barrier can be detected only when it is within the sensing range of at least one sensor. The sensor network establishes barrier coverage of the segment if no intruder can penetrate the barrier from any direction in the plane without being detected. Starting from arbitrary initial positions of sensors on the line they are interested in finnding final positions of sensors that establish barrier coverage and minimize the maximum distance traversed by any sensor. They have distinguished several variants of the problem, based on (a) whether or not the sensors have identical ranges, (b) whether

or not complete coverage is possible and (c) in the case when complete coverage is impossible, whether or not the maximal coverage is required to be contiguous. For the case of n sensors with identical range, when complete coverage is impossible, they have given linear time optimal algorithms that achieve maximal coverage, both for the contiguous and non-contiguous case. When complete coverage is possible, they give an $O(n^2)$ time algorithm for an optimal solution, a linear time approximation scheme with approximation factor 2, and a $(1 + \varepsilon)$ PTAS. When the sensors have unequal ranges we show that a variation of the problem is NP-complete and identify some instances which can be solved with our algorithms for sensors with unequal ranges.

Chapter 3

Problem Definitions and Preliminaries

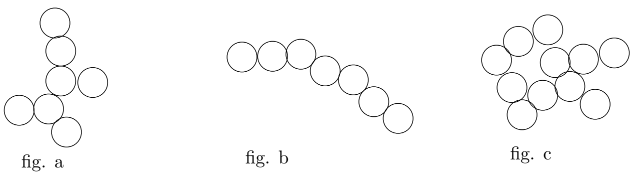
3.1 Linear

Let $L = [0, \ell]$ be a line segment on the x-axis. A set $D = \{d_1, d_2, \ldots, d_n\}$ of n discs are initially located on the x-axis. Each disc $d_i \in D$ has a range $r > 0$ and centred at x_i, for all $i = 1, \ldots, n$. We assume $x_1 \leq x_2 \leq \ldots \leq x_n$. $[x_i - r_i, x_i + r_i]$ is the covering interval of d_i. We need to find a set of destinations on the x-axis, $\{y_1, y_2, \ldots, y_n\}$, for the discs such that the discs are in a connected configuration and the maximum movement (i.e.,$\max_{1 \leq i \leq n}\{|x_i - y_i|\}$) made by the disc is minimized.

3.2 2D

We have a set $D = \{d_1, d_2, \ldots, d_n\}$ of n discs of radius r which are placed in a 2-D plane. Each disc $d_i \in D$ has its centre at the co-ordinate (x_i, y_i) and it covers the region enclosed in the disc. We are interested in finding a set of destinations on the 2-D plane ,$\{(p_1, q_1), (p_2, q_2), \ldots (p_n, q_n)\}$, for the discs so that they are in a connected configuration and the maximum movement (i.e.,$\max_{1 \leq i \leq n}\{\sqrt{(p_i - x_i)^2 + (q_i - y_i)^2}\}$) made by disc is minimized.

Definition 3.2.1 *Connected configuration : We say two discs to be connected when they touch each other. Consider each disc as a vertex in a graph put an edge when two discs are connected. Starting from any arbitrary vertex if there exists a path to each and every other vertex we say the disc to be in a connected configuration.*



Some examples of connected configuration.

3.2.1 Upper bound for Opt

We get a Minimum Enclosing Circle(MEC) which encloses the set of n discs as in figure 3.1. Say C is the centre of the MEC and all the discs are moved towards the centre.The radius of the MEC computed would be the upper bound for the maximum movement of the discs.

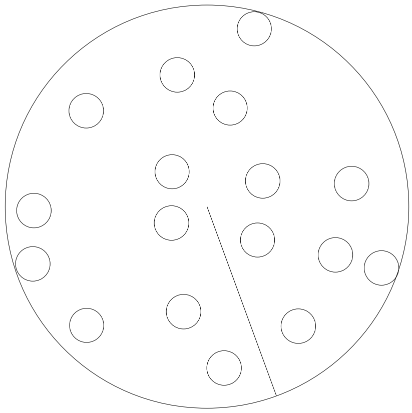


Figure 3.1: Minimum Enclosing Circle enclosing the discs in 2D plane.

Observation 3.2.2 *It is not necessary that discs move towards the centre of the MEC.*

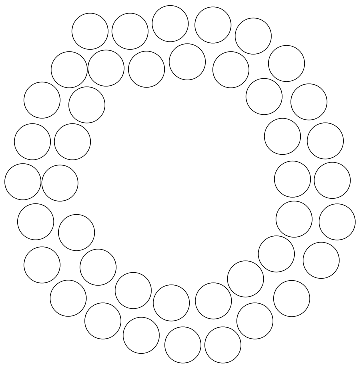


Figure 3.2: Discs may not have to move towards the centre of MEC.

3.2.2 3 disc

Here we have 3 discs of same radius which are moved to form connected configuration.

Observation 3.2.3 *Either three discs moves the same distance or two moves same and one moves lesser.*

Observation 3.2.4 *When the triangle is obtuse two points lying on the larger side move the same optimum distance and the other less (for circles with less radius). If the radius of the circle is large than the* 3 *points move the same optimum distance.*

Theorem 3.2.5 *In acute-angled triangle we always have a solution where the three move the same optimum distance and are in connected configuration.*

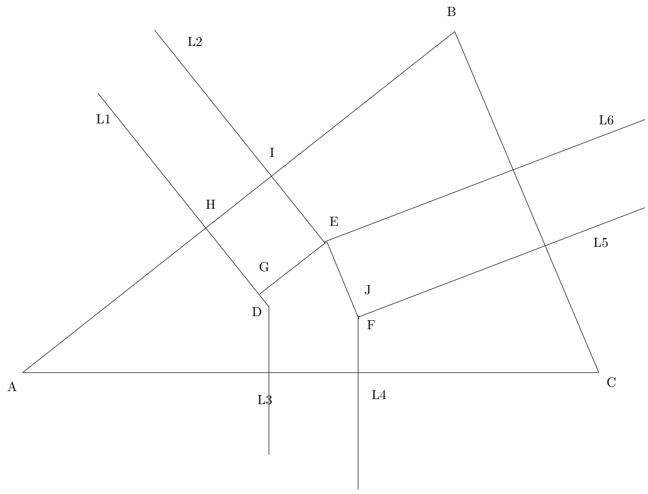


Figure 3.3: Acute angled triangle showing that three discs on the vertices of the triangle move equally to minimize the maximum movement

Proof : As in figure 3.2 we name the points of the triangle as A, B and C where the discs of same radius r lie. Now we draw two lines($L1$ and $L2$) parallel to the perpendicular bisector of A, B one towards A and other towards B. Both the lines are r unit from the midpoint of AB. Similarly parallel lines($L3, L4, L5$ and $L6$) are drawn on the side BC and CA as shown in fig 3.2. These lines meet at point D, E and F. The length of AD, BE and CF is computed and minimum of them is found (say it is BE). The circle at B is placed at E. The circles at A and C is moved to touch the circle at E by moving them distance equal to BE. This is done as follows, from E a line parallel to AB is drawn to the line $L1$ say at G. To this point G circle at A is brought. Since AHG is congruent to $BIE, AG = BE$. Similarly a point J in line $L5$ is found such that $CJ = BE$. The circle at C is brought at J. We claim that this is the optimum.

3.2.3 4 disc

Four discs of same radius on the vertices of a rectangle, square, parallelogram rhombus were studied. It was found that the final configuration of the discs in which the maximum movement is minimized depends on the diagonals of the above mentioned geometry. We start with same diagonals i.e. a square, we would see that the target configuration would look like configuration I as shown in figure 3.3. When this square is stretched such that it remains a parallelogram and one of of its diagonal length increases and the other decreases the target configuration changes from configuration I to configuration II and finally remains in configuration III.

3.2.4 Circle problem

The given n discs were placed on a larger circle(LC) such that the centre of the n discs lie on the circumference of the LC. Now the discs are moved to form a connected configuration such that maximum movement

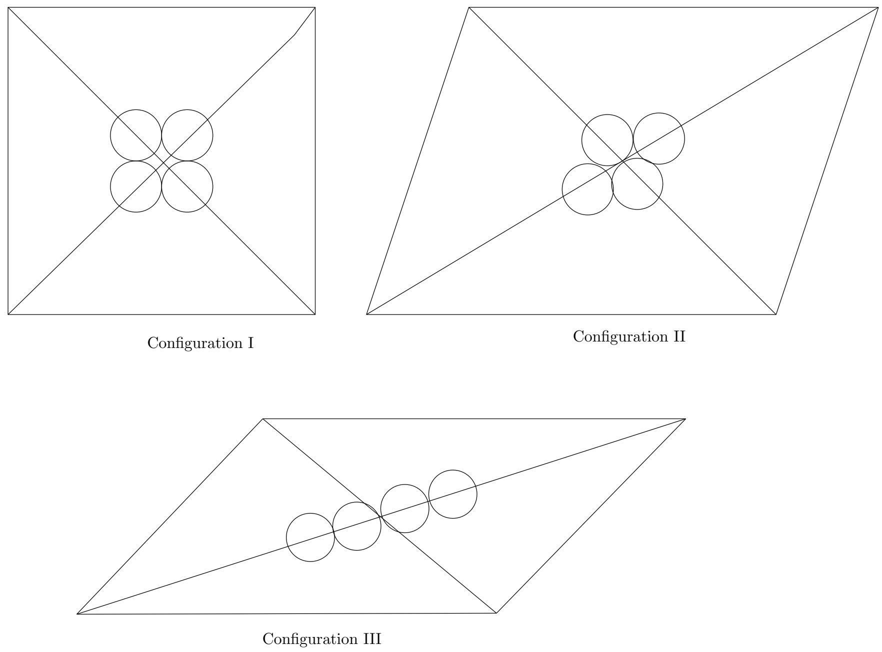


Figure 3.4: Transition of final configuration as the diagonals change

is minimized. Also a smaller concentric circle to the LC is placed and the discs placed in the larger circle are moved to the smaller circle with the same objective, i.e. minimizing the maximum movement to get connected.

3.3 Tree

Discs or pebbles that are n in number are placed on the vertices of a tree, the number of vertices is more than n. Only single pebble is allowed to be placed in a vertex of the tree at a given time. We need to move the pebbles along edges to a connected configuration, i.e. the induced subgraph of the vertices that has pebble is a connected component. Erik Demaine et.al.[Demaine09] have given a polynomial time solution for ConMax in case of trees using bipartite matching.

Definition 3.3.1 *Conglomerate : A tree where each vertex has a pebble.*

Definition 3.3.2 *Pruned tree : It is a tree which results by deleting leaf nodes which do not have pebbles. So a pruned tree has pebbles in all of its leaf nodes.*

Definition 3.3.3 *Leaf Conglomerate: It is a conglomerate which has exactly one vertex(V) which has exactly one non-pebble neighbour in the pruned tree. V is then the root of the leaf conglomerate.*

Definition 3.3.4 *Junction Conglomerate: A conglomerte which has a vertex which has more than one non-pebble neighbour or at least two vertices with one non-pebble neighbour.*

Definition 3.3.5 *Skeleton: Let V be the set of vertices in the evolving tree after every leaf conglomerate has path min-max $= k$, L be the set of vertices in the leaf conglomerates. The skeleton is a sub-tree of the evolving tree whose vertex set is $W_1 \subset V - L$ and each vertex $w \in W_1$ lie on the unique path between a pair of leaf conglomerates.*

3.3.1 Formation and movement of pebbles in a conglomerate

We have defined that a conglomerate is a tree where each vertex has a pebble. Initially we get a pruned tree, where all the leaf vertices have pebble. Say we are going to check that in k movements we have a connected configuration or not. We would first move the leaf pebble or leaf conglomerate k steps then move the pebbles in junction conglomerate.

We allow only leaf pebbles to move at first, they can move only along one direction or edge. As we get a pruned tree we may have formation of leaf conglomerates already existing or it may form as the leaf pebbles move in our evolving tree. Some of the leaf conglomerates may become junction conglomerate. So now the question is when leaf conglomerate is moved which path in the conglomerate moves. For a given leaf conglomerate we would keep track of maximum movement made in all the paths. The path which has minimum of the maximum movement, min-max, is chosen for movement. Now, when we have used up our k step movement in all the leaf conglomerates. We would now use pebbles in junction conglomerate for achieving connectivity.

Chapter 4

Linear problem

Without loss of generality we would consider the discs to be of unit radius.

Observation 4.0.6 *The two extreme discs would move the same distance and that would be opt.*

4.1 Algorithm for linear problem

Step 1 : Find the two extreme discs in the left and right of the given line/bar.
Step 2 : Get the number of discs between them.
Step 3 : Add up the gaps between discs, say it comes out G units, assuming no overlapping discs.
Step 4 : Now compute the distance that the leftmost and rightmost disc needs to move, it would be moving half of G.
Step 5 : We start calculating the position the disc would move to from left. The leftmost disc would move right by $G/2$ units. As we go right we compute *gap* to the left of the disc to be moved.
Compute $m = G/2$- *gap*. If m is positive than we need to move it right by m units else we move it to left by m units.

We claim that in the final connected configuration where the maximum movement would be minimized, the extreme discs would be moving equally and $G/2$ units. We also claim that these two points would move the most.
This algorithm would take $O(nlogn)$ time as sorting is required for knowing the ordering of the discs.

Lemma 4.1.1 *The extreme discs would move $G/2$ units which is optimum to form a connected configuration.*

Proof : Let us assume $G/2$ movement of extreme discs is not optimum ,i.e., there exist a movement x less than $G/2$ of extreme discs which is optimum. After x momvement of extreme discs, there exist atleast two discs which have a gap so our discs are disconnected.

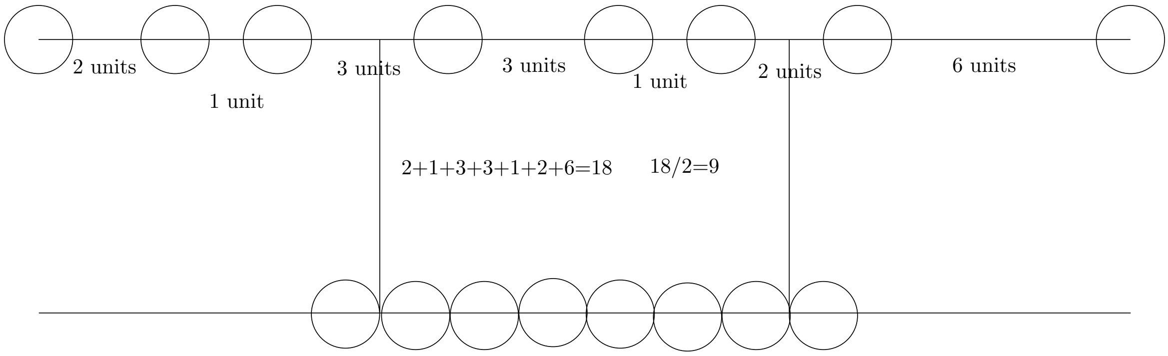


Figure 4.1: Extreme left and right discs move equal and opt units

Idea for an efficient solution

We are focussing on finding a point which we name as focus, such that all the discs to the left and right of focus would move towards it. So no ordering is required and $O(n)$ time solution is possible. We have a line $L = [0, r]$ be a line segment on the x-axis. Let C be the centre, i.e. a point at $r/2$. Leaving the extreme discs, without loss of generality let n_1 discs be there to the left of centre and $n_1 + \delta$ discs to the right of it.

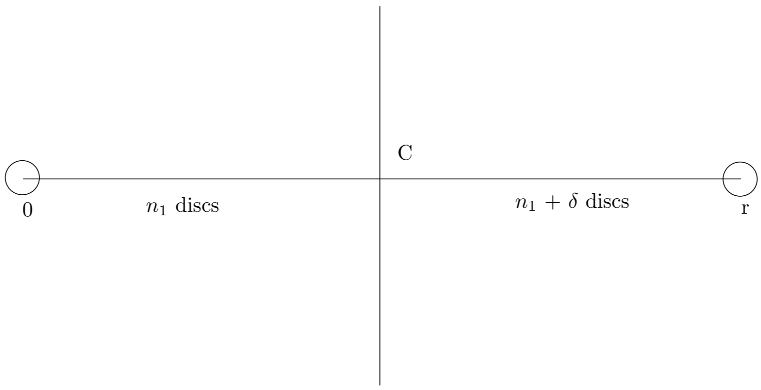


Figure 4.2: Idea for an efficient solution

If $\delta = 0$, all the discs to the left and right of centre will move towards the centre and get connected. Now, when δ is greater than zero let all the discs move towards the centre and get connected. The discs in the right of centre could have helped discs in the left.

Chapter 5

Tree problem

Observation 5.0.2 *There will always be atleast two leaf conglomerates who has only one direction to move one at a time.*

Erik Demaine et.al.[Demaine09] have given a polynomial time solution for ConMax in case of trees using bipartite matching.

Theorem 5.0.3 *Given a tree T and a configuration of k pebbles on T , ConMax can be solved in polynomial time.*

Proof : First we guess a vertex v of T that is occupied by a pebble in the target configuration. Second we guess the maximum movement k, $0 \leq k \leq n$ required by the optimal solution.
We compute a subset of vertices of T that must be occupied by pebbles in the target configuration. For each pebble in the initial configuration, consider moving it toward v by up to k steps (stopping if it reaches v). This vertex x is the closest the pebble could get to v in any solution with a maximum movement of k. Thus, every vertex along the path from x to v (including the endpoints) must be occupied by a pebble in the target configuration. We call these vertices forced.
To determine whether all forced vertices can indeed be occupied by pebbles (and thus whether this guess of v and k is valid), we use bipartite matching. Define the bipartite graph $H = (W1, W2, F)$ where $W1$ is the set of pebbles, $W2$ is the set of forced vertices, and edges in F connect a pebble to every forced vertex that is within distance k in G. A maximum-cardinality matching of this graph H covers every vertex in $W2$ if and only if the pebbles can be moved to occupy the forced vertices.
Any extra pebbles not used in the matching can be moved to an arbitrary forced vertex: every pebble can be moved to some forced vertex, namely x. The forced vertices induce a connected subgraph of G, so we obtain a solution to ConMax with maximum movement k if this is possible.

Observation 5.0.4 *It is very significant to decide as discs in which path of the conglomerate needs to be moved, so junction conglomerate play an important role to minimize the maximum movement.*

In the given figure we have leaf conglomerates at $A1, K, X$. Junction conglomerates are placed in $A4, A9, B, D, R, G$. In this scenario the minimized maximum movement would be 2. The leaf conglomerate at $A1, K$,and X would move to the vertex at $A3, I$ and V respectively. Leaf conglomerates at $A1$ and

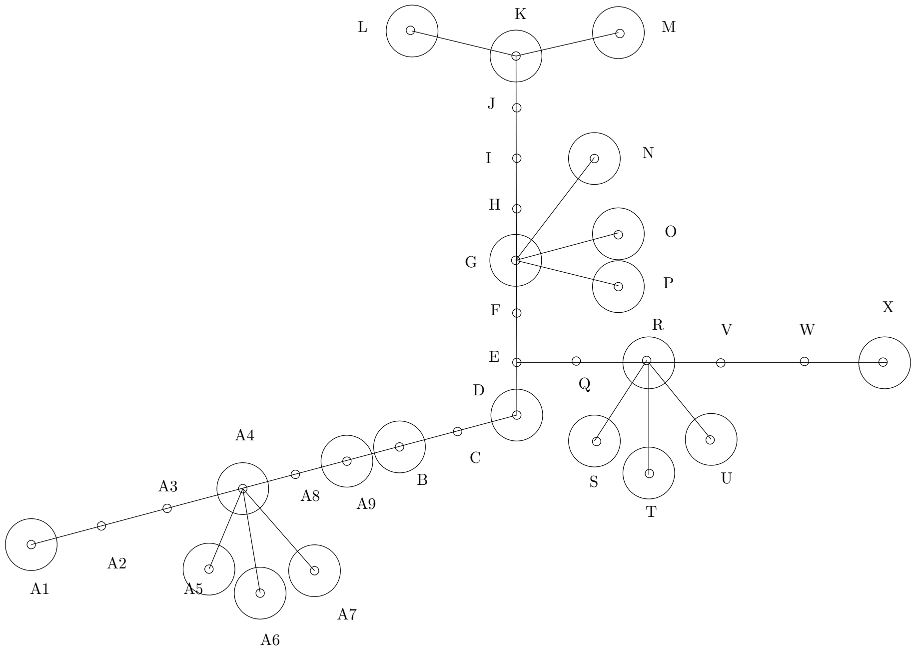


Figure 5.1: Role played by junction conglomerate

X would form a new leaf conglomerate with the junction conglomerate at $A4$ and R respectively.
The new leaf conglomerate at $A4$ has its root whose movement is zero. Pebble at $A4$ moves to $A8$ and pebble at $A7$ moves to $A4$, this way $A9$ and B also get connected to it. Now pebble at $B, A9, A8, A4$ and $A6$ moves to $C, B, A9, A8, A4$ respectively and connects to D.Till now the movement has been only two. Similarly the leaf conglomerate at X moves to junction conglomerate at R and the pebbles in junction conglomerate move to E in two movements.
The leaf conglomerate at k has come to I, the pebble at G, N moves to H, G respectively. Now the pebble at G, O moves to F, G respectively. The junction conglomerate at G plays an important role by connecting leaf conglomerate at I and also connecting other connected component at E. All this happens in two movements, which is the minimum movement possible to achieve connectivity. It is clear from the figure that connectivity cannot be achieved in a single movement.

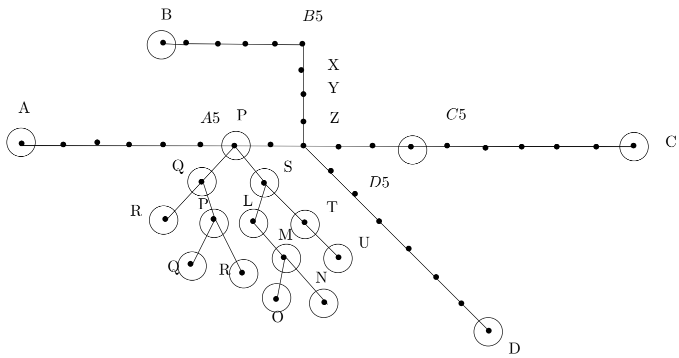


Figure 5.2: Importance of which path to choose for pebble movement.

5.1 Algorithm for tree problem

We give a better algorithm, treeConMax, in the tree case which is as follows:
(1) We prune off the leaf vertices not containing pebbles until we have all leaf vertices with pebbles,i.e. we have a pruned tree, set $k = 1$.
(2) We move the pebbles in the leaf conglomerate k steps, if it is connected we are done else the pebbles in the junction conglomerate are moved in a path which makes it move further atmost k steps.
(3) If it is not connected yet than we double the value of k and goto step 2 else we know the last value of k(say k_l) in which it was disconnected and the value of k (say k_n) which just made it connected. Between these values of k we apply binary search method to get new k(say k_o) we keep going to step 2 with the new value of k_o. This way we get the value of least k value which is making it connected.

5.1.1 Shortcomings of the Algorithm

We have 11 pebbles $a, b, \ldots, k$ in the initial configuration as shown in $figure$ 5.3(a). In the figure $figure$ 5.3(b) we see that a, h has moved 4 edges. The pebble/disc g, j, k moves 4 edges and touches h. The pebble b moves 2 edges and connects with k. The pebble/disc f, e moves 4 edges to connect with b and i moves 4 edges to connect with f. So we have a connected configuration. Here we have moved the junction conglomerate formed by the pebbles g, j and k furthest.

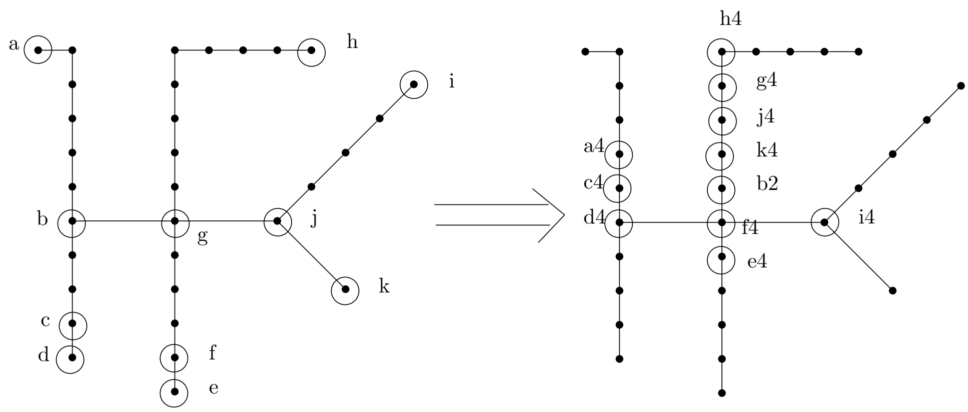


fig (a) fig (b)

Figure 5.3: Example showing furthest movement of junction conglomerate

We have an example where moving junction conglomerate furthest may not be a better choice and is shown in $figure$ 5.4. Here the pebble h in the junction conglomerate does not move furthest.

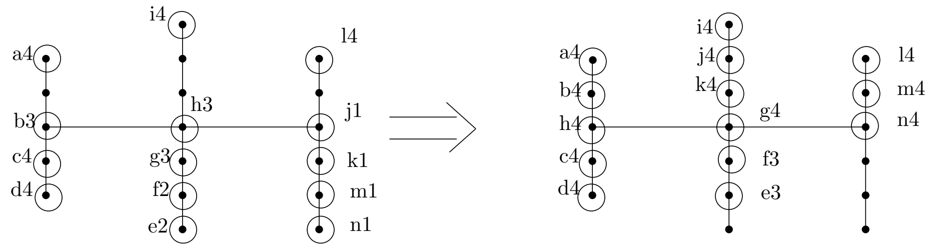


Figure 5.4: Example showing furthest movement may not help sometimes

Chapter 6

Conclusion and Future Works

In this work we have focussed on minimizing the maximum movement for connectivity problem, 3 points and 4 points example have been seen but the general problem for n points is yet to be solved in 2-dimension. Discs on a line have been solved. The study on circle case is incomplete. The other variants of the problem like minimizing the number of movements and total movement to achieve connectivity also will be studied in the coming days. In this work we have looked at discs with same radius only and we have not taken overlapping discs. Problems with different radius and overlapping discs would also be significant.

Bibliography

[Demaine09] Erik D. Demaine, MohammadTaghi Hajiaghayi, Hamid Mahini, Shayan Oveis Gharan,Amin Sayedi-Roshkhar and Morteza Zadimoghaddam. Minimizing movement. ACM Transactions on Algorithms, 5(3), Article 30, July 2009. Preliminary version appeared at SODA 2007.

[Berman] Piotr Berman, Erik D. Demaine, and Morteza Zadimoghaddam, O(1)-Approximations for Maximum Movement Problems

[Demaine009] Erik D. Demaine, MohammadTaghi Hajiaghayi, and Daniel Marx. Minimizing Movement: Fixed-Parameter Tractability. In Proceedings of the 17th Annual European Symposium on Algorithms (ESA 2009), pages 718-729.

[Friggstad08] Zachary Friggstad and Mohammad R. Salavatipour. Minimizing movement in mobile facility location problems. In Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2008), pages 357366.

[Chen] Danny Z. Chen, Yan Gu, Jian Li, Haitao Wang. Algorithms on Minimizing the Maximum Sensor Movement for Barrier Coverage of a Linear Domain.

[Czyzowicz] J. Czyzowicz, E. Kranakis, D. Krizanc, I. Lambadaris, L. Narayanan, J.Opatrny, L. Stacho, J. Urrutia, and M. Yazdani. On Minimizing the Maximum Sensor Movement for Barrier Coverage of a Line Segment.