University of Alberta

Graph Clustering: Complexity, Sequential and Parallel Algorithms

by

Nesrine Abbas

Technical Report TR 95–01
February 1995

DEPARTMENT OF COMPUTING SCIENCE
The University of Alberta
Edmonton, Alberta, Canada

Graph Clustering: Complexity, Sequential and Parallel Algorithms

Nesrine Abbas
Department of Computing Science
University of Alberta
Edmonton, Alberta, T6G 2H1
CANADA

Abstract

In this thesis we study graph clustering on two well known families of graphs that arise in many applications, namely bipartite and chordal graphs. We study two specific types of clustering problems. On the one hand we seek a partition of the vertex set of a graph into a bounded number of sets so that a prespecified function determined by the diameters of the induced subgraphs is minimized. On the other hand we seek one subset of the vertex set that has a fixed number of vertices and induces the maximum possible number of edges.

The first problem type constitutes the main theme of this thesis. We define a class of functions on partitions of vertices, which we call *monotone diameter functions*, that will enable us to group many problems into one general problem. Examples of these problems are the following: partition the vertices of a graph into a limited number of subgraphs of bounded diameter, and partition the vertices of a graph into a limited number of subgraphs so that the sum of the diameters of the subgraphs is bounded.

We prove that the problem of partitioning the vertices of a graph into subgraphs of bounded diameter is *NP*-complete on bipartite and chordal graphs. In contrast with these negative results, we give linear time sequential algorithms for the same problem on bipartite permutation and interval graphs, the first being a subset of bipartite graphs and the second being a subset of chordal graphs.

We also present efficient parallel algorithms for the general problem on biconvex and interval graphs.

During the course of studying this problem on biconvex graphs, we prove that biconvex graphs have a certain ordering of the vertices, which we call *a biconvex straight ordering*, that has promising algorithmic potential. We present an efficient parallel algorithm for constructing a biconvex straight ordering for any biconvex graph.

We also study the problem of finding a cluster of bounded size with the maximum possible number of edges, and present an efficient parallel algorithm for solving this problem on a class of unit interval graphs.

Acknowledgements

To God I owe everything. To everyone that helped make the completion of this thesis possible I offer my thanks.

My supervisor, Lorna Stewart, provided me with expert guidance, many hours of discussion, and financial support to complete the thesis and to attend conferences. I am grateful for her time, effort, patience, and support.

I am thankful to the other members of my committee: Stephen Hedetniemi, Jim Hoover, Joseph Culberson, and Andy Liu. Special thanks to Stephen Hedetniemi for suggesting many thoughtful improvements, to Jim Hoover for the time he spent with me discussing some of the parallel algorithms in this thesis, to Joseph Culberson for carefully reading the thesis, and to Andy Liu for pointing out a shorter proof for one of the theorems.

To my parents, I am indebted with gratitude for their infinite support throughout all stages of the preparation of this thesis, and throughout all stages of my life.

I am thankful to my husband, Ehab, for offering me invaluable advice on thesis writing and for his patience and support. To my children, Aser and Salma, I am grateful for their love, smiles, laughter, and hugs.

Contents

Chapter 1

Introduction

1.1 Overview

The last few decades have witnessed the wide use of graphs as an effective modeling tool for many scientific problems. In this thesis we study a class of graph problems that have strong links to clustering problems. Clustering calls for partitioning a set of objects into clusters, or subsets, so that 'similar' objects are in the same cluster and 'different' objects are in different clusters. A definite meaning is attached to a cluster when devising an algorithm.

The clustering problems considered in this thesis seek a partition that minimizes a specific function determined by the diameters of the clusters in the partition. For example we might be interested in a partition where each set has a bounded diameter. Another related problem, also considered in this thesis, seeks a cluster of bounded size that has the maximum possible number of edges.

Clustering methods are needed in many areas. Linguistics, image processing, pattern recognition, and VLSI design are examples of such application areas. We elaborate on some specific clustering problems in these areas in the next section.

From a graph-theoretic point of view we follow a few research directions, that are part of the general trend in this area. These directions follow.

1. Characterize new classes of problems that encapsulate older problems. This direction makes most sense if the new class can be solved by efficient algorithms.

2. Classify problems on general and special graphs according to their computational complexity.

3. Devising sequential and parallel efficient exact algorithms for problems on general and special graphs.

In the first of these directions we define a problem that generalizes the problem of partitioning vertices of a graph into a limited number of subgraphs with bounded diameter. That problem, in turn, encompasses the problem of partitioning vertices of a graph into cliques. The proposed generalization is based on the idea of using *monotone diameter functions* defined on partitions of graphs. We give the formal definitions for these functions and the general problem in section 1.3.

In the second direction we give *NP*-completeness results for the problem of partitioning vertices into a limited number of subgraphs with bounded diameter on bipartite and chordal graphs. These results, in effect, imply that a polynomial time algorithm for these graph families is most probably not feasible.

The results of the above direction lead us to the third research direction. We consider biconvex and interval graphs, which are subsets of bipartite and chordal graphs, respectively, and devise sequential and parallel algorithms for clustering problems on these graphs. The positive results along this direction are fruitful matchings between the mathematical properties of the source problems and the structural properties of the target classes of graphs. Along this direction, we introduce a new characterization of biconvex graphs that has great algorithmic potential. One of the implications of the new characterization is that shortest paths in biconvex graphs have special geometric properties. Another implication is that the neighborhood sets of all vertices are contained in a bipartite permutation induced subgraph. These properties have made our task of designing an algorithm for biconvex graphs much easier.

The graphs we consider in this thesis arise in numerous applications. Bipartite graphs arise in scheduling and job-processor problems, where generally it is required to match a vertex or a set of vertices in one partition with a vertex or a set of vertices in the other partition. For example, in [23] and [24] convex graphs are used to model several scheduling problems. Biconvex graphs are used to model industrial problems, as in [38]. Bipartite graphs arise in image processing applications, where nodes may represent pixels in an image, and two nodes are joined by an edge if the respective nodes are horizontal or vertical neighbors and are 'close enough' to each other.

Chordal graphs have applications in Gaussian elimination [69] and databases [7].

Interval graphs are used to model many problems in genetics, archeology, and sociology (for example see [68]). They also arise in scheduling ([23] and [24]), and VLSI ([23] and [40]) problems. They are mostly used to model temporal events, where each interval corresponds to one such event.

Moreover, biconvex and interval graphs are algorithmically very appealing since they have certain orderings of the vertices that facilitate the design of polynomial algorithms.

We now give an overview of the remaining part of this chapter. In the next section we discuss, in some detail, graph-theoretic clustering problems in three application areas. We then survey, in the following section, some results on clustering problems when the objective function is diameter related; the main problem in this thesis is one of these problems. The fourth section of this chapter contains formal statements of the two problems considered in this thesis. Previous results on these problems on special graphs are presented in section 1.5, together with a brief statement of our contributions. We end this chapter with a description of the remainder of this thesis.

For any graph theory terms, the reader is referred to appendix A. Appendix B contains definitions of graph families. Any other graph definitions may be found in [40]. Appendix C contains definitions of complexity classes, complete problems, and the parallel computation model that we use in this thesis. Other definitions in that respect may be found in [33], [44], and [36]. Throughout this thesis the size of the input to any problem will be considered $n+m$, where n is the number of vertices and m is the number of edges in the input graph. We will use the term *efficient sequential algorithm* to refer to sequential algorithms that run in time polynomial in $n + m$. The term *efficient parallel algorithm* will refer to parallel algorithms that use a number of processors polynomial in $n + m$, and run in time polylogarithmic in $n + m$. We will often say that problems that have efficient parallel algorithms belong to the class *NC*. However some of these problems would be more accurately described as belonging to the class *FNC* [44]. Since the distinctions are always apparent, we trust the reader will realize them. The parallel computation model that we use is the *Concurrent Read Exclusive Write Parallel Random Access Machine*, abbreviated *CREW-PRAM* (see appendix C for definitions).

1.2 Applications of Clustering

We consider three areas in which clustering problems emerge. Our aim is to reemphasize the importance and wide applicability of graph partitioning problems.

(1) Information Retrieval: Gotlieb and Kumar [43] use graph-theoretic clustering methods for the reorganization of indexing terms. Vertices in a graph represent indexing terms and edges exist between two vertices if the similarity between the corresponding pair of terms is greater than a certain threshold. The similarity is represented as an edge-weight, and is computed from some association measure. To group terms into concepts or clusters the author compares two methods.

In the first method clusters correspond to connected components of the graph. This method has the drawback that clusters may contain long paths of terms between unrelated vertices, e.g. a path between two terms that is equal to the diameter of the cluster.

In the second method clusters correspond to maximal complete graphs. The drawback in this method is the large number of maximal cliques that have to be considered, and the possibility of having overlapping cliques. The latter drawback is overcome by the authors by merging overlapping cliques into one cluster.

It should be noted that a clustering method that produces clusters with bounded diameter would be a compromise between the above methods. In such a method a connected component may be broken into more than one cluster, and thus paths between unrelated vertices will be of bounded length. In addition, a cluster may contain more than one clique (though not necessarily maximal).

(2) Image Segmentation: A graph may be used to model an image by assigning vertices to pixels in a one-to-one fashion, and assigning edges between pixels if they obey a certain neighboring condition. For example edges may exist between two vertices if the corresponding pixels are vertical or horizontal neighbors, giving rise to a bipartite graph that is a partial grid.

In [76] a graph-theoretic clustering method is used for image segmentation. Patches of an image are in one-to-one correspondence with vertices of the graph. Edges exist between vertices if the corresponding patches are neighbors. Weights are assigned to edges to reflect that connectivity. A spanning tree is then found using a maximal flow algorithm, where edge weights in the tree correspond to the value of the maximum flow between the endpoints. From that tree, edges are removed to obtain the desired clusters. Effectively the method produces clusters for which the sum of inter-cluster edge-weights is minimized.

Another example in that direction is the work done in [50]. The problem addressed in that paper is that of matching two images of the same scene from different viewing positions. A graph is built, where each vertex $v(i, j)$ corresponds to a matching of feature i in the first image with feature j in the second image. Edges in that graph represent compatible assignments. A 'best' matching is found by searching for a 'large' maximal clique that optimizes a certain objective function.

(3) VLSI Design: Consider the following problem. It is required to allocate vertices of a circuit to distinct boards, so that the number of boards is bounded and a certain constraint on the partition is met. For example, the area of the board might limit the number of vertices in each board, or the number of connectors might limit the number of external connections to the vertices on the board. This problem is studied in [65]. Some *NP*-completeness results are reported and a pseudo-polynomial dynamic program is given for the case when the input graph is a tree.

1.3 Diameter Based Clustering Criteria

In a graph-theoretic context a cluster may be viewed in several ways. It may be a clique (e.g. [43] by Gotlieb and Kumar), a subgraph with a maximum number of edges (e.g. [21] by Corneil and Perl), a subgraph with bounded diameter (e.g. [31] by Farley, Hedetniemi, and

Proskurowski), a bond (e.g. [59] by Matula and Beck), or a connected component (e.g. [58] by Matula, and [43] by Gotlieb and Kumar). For definitions of those terms see appendix A. The clustering criteria we review in this section are based on the diameter, since the main problem of this thesis falls in this category. We review results on two clustering problems: one requires a partition where each set has a bounded diameter, and the other requires a partition with a bounded sum of cluster diameters. Both problems are studied in this thesis (as part of a more general problem).

Although clustering algorithms where the objective is to optimize a function of the diameters of the clusters have been extensively studied in the literature (e.g. [17] and [61]), few papers deal with these problems on special graphs. Most of the work reported is on weighted complete graphs, where weights represent distances between vertices. Surely, the intractability of a certain problem on the weighted complete graph opens the door to research on special graphs. Unfortunately, the solution of a problem on weighted complete graphs does not readily extend to a solution on special graphs. For example, see figure 1.1 where it is required to bipartition the graph G (figure 1.1 a) so each set has diameter smaller than three. Figure 1.1 b shows the weighted complete graph, G_c, obtained from G by calculating the distances between all pairs of vertices, and adding edges carrying these distances as weights. Figure 1.1 c shows an optimal partition for G_c, which does not correspond to a valid partition for G.

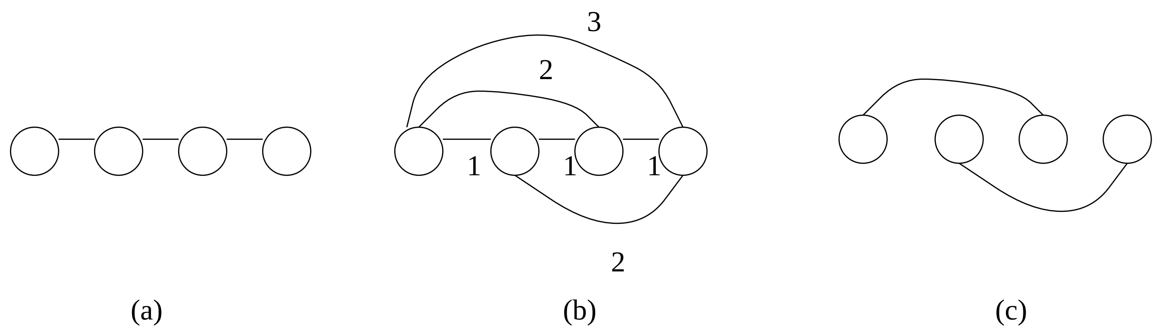

Figure 1.1: (a) A graph G
(b) Weighted complete graph G_c
(c) An optimal partition for G_c but not for G

Before we go any further, some definitions are due.

Let O be the object set to be clustered. The input to such a clustering problem may be a distance matrix D. Each entry in D, $d(i,j)$, is the distance between objects o_i and o_j, where $d(i,j)$ is a non-negative integer, $d(i,j) = d(j,i)$, and $d(i,i) = 0$.

Let $\mathcal{Z} = \{Z_1, Z_2, ...Z_k\}$ be a partition of O into k disjoint clusters.

Define the *diameter of a cluster* Z, $diam(Z)$, to be the maximum distance between any pair of objects in Z, $diam(Z) = \max_{o_i, oj \in Z} d(i,j)$.

The *diameter of a partition* $\mathcal{Z}$, $diam(\mathcal{Z}) = \max_i\{diam(Z_i) : Z_i \in \mathcal{Z}\}$, is the maximum of its cluster diameters.

The problem of whether or not there is a partition with a minimum diameter is *NP*-complete when k (the number of clusters in the partition) is greater than two as shown in [25], [17], [47], and [75].

In fact Welch [75] proves a more general partitioning problem to be *NP*-complete where the objective function is always non-negative and is equal to zero if and only if all within cluster distances are zero. An example of such an objective function is $diam(\mathcal{Z})$.

The problem of whether or not there is a partition of cardinality two with minimum diameter is polynomially solved in [67], [17], and [61]. Rao [67] uses mathematical programming. In [61] Monma and Suri show that an optimal bipartition may be obtained by coloring a maximum spanning tree of the input graph; each cluster corresponds to the vertices in one of the two color classes. Brucker [17] uses a similar approach.

A branch and bound algorithm is presented in [25] and [47] to solve the minimum diameter partition problem for any k.

The problem of whether or not there is a cardinality k partition with the minimum sum of cluster diameters is *NP*-complete when k is greater than two, as shown in [75] and [17].

This problem has a polynomial solution when $k = 2$, as shown in [48] and [61].

Other objective functions of distances have been considered. Some of the results are summarized in table 1.1. In this table we include results on clustering criteria based on the diameter or the *split*, which is a measure related to the diameter. The *split of a cluster* Z, $split(Z)$, is defined to be the minimum distance between an object in Z and an object outside of it, $split(Z) = \min_{o_i \in Z, o_j \notin Z} d(i,j)$. The *split of a partition* $\mathcal{Z}$, $split(\mathcal{Z})$, is equal to the minimum of its cluster splits. The diameter and the split are two of the most commonly used clustering criteria.

Objective	*Results*
Partition with minimum diameter	$k = 2$ *Polynomial* [67], [17], and [61] $k > 2$ *NP-complete* [75], [25], [17], and [47]
Partition with minimum sum of cluster diameters	$k = 2$ *Polynomial* [48], and [61] $k > 2$ *NP-complete* [75] and [17]
Partition with maximum split	*Polynomial* [25]
Partition with maximum sum of cluster splits	*Polynomial* [49]
Partition with minimum sum of average squared within-cluster distances	$k > 2$ *NP-complete* [75]
Partition with minimum sum of within cluster distances	$k = 2$ *NP-complete* [17] $k > 2$ *NP-complete* [75] and [17]
Partition with minimum sum of average cluster distances	$k = 2$ *NP-complete* [17]

Table 1.1: Some Clustering Problems and Results

1.4 Statement of Problems

It is clear that to partition N objects into k non-empty sets, complete enumeration is very time consuming. Also as noted in the above paragraphs, many clustering problems are computationally hard. For these reasons researchers have either devised approximate algorithms (e.g. [61]), used branch and bound techniques (e.g. [57], [25], and [47]), integer and quadratic programming (e.g. [67]), dynamic programming (e.g. [51]), or restricted the domain of the input to the problem.

For example the input may be restricted to points in Euclidean space, or to special graphs. Examples of work done in the former direction include the following. The problems in [3], in [4], and [61] where bipartitioning points to minimize the maximum diameter is the objective. Brucker [17] studies, among other problems, bipartitioning Euclidean points to minimize the sum of the average cluster distances. In [13] clustering points to maximize the sum of distances between objects in different clusters is studied.

Another alternative is to restrict the input to special graphs with practical applicability and favorable algorithmic properties. We employ that approach.

Our aim is to explore the area where the complexity of the chosen clustering problem is yet to be determined. In that respect two directions are pursued. In one direction we show the *NP*-completeness of the problem on the selected graph family. In the other direction we try to narrow the gap between the polynomially solved instances of the problem and the computationally hard instances. Along that path we give efficient algorithms (*NC* and linear) for our clustering problems on special graphs.

In this thesis we assume that the input to the clustering problem is an unweighted graph $G = (V, E)$, with vertex set V and edge set E.

The solution to the first problem that we study is a partition of the vertex set into a bounded number of sets, whereas the solution of the second problem is a bounded size subset of the vertex set.

In the first problem we make use of a set of *monotone diameter functions* which we define on partitions of graphs, to generalize some well known problems. We now define that class of functions.

Let $G = (V, E)$ be a graph. Let $< W >$ be the subgraph induced by a set $W \subseteq V$. Let $diam(W)$ denote the diameter of $< W >$. For the purpose of the next definition, we shall use the term *subpartition*, $\mathcal{Z}_a$, of G, to mean a partition of the subgraph, $< V_a >$, of G induced by the set $V_a \subseteq V$.

Definition 1.1: A function f from the set of all subpartitions of G to the set of non-negative integers, is said to be *a monotone diameter function*, if it satisfies the following.

For any subpartition $\mathcal{Z}$ of G

$f(\mathcal{Z}) = \bullet(diam(Z_1), diam(Z_2), \ldots diam(Z_{|\mathcal{Z}|}))$, where $\bullet$ is an associative operator.

For any subpartitions $\mathcal{Z}_a$, $\mathcal{Z}_b$, and $\mathcal{Z}_c$ of G that satisfy $V_c \bigcap V_a = \phi$ and $V_c \bigcap V_b = \phi$

$f(\mathcal{Z}_a) \leq f(\mathcal{Z}_b)$ implies $f(\mathcal{Z}_a \bigcup \mathcal{Z}_c) \leq f(\mathcal{Z}_b \bigcup \mathcal{Z}_c)$.

Now we define the first problem.

Problem *PVMF*: Partitioning Vertices to Minimize Monotone Function: Given $G = (V, E)$, monotone diameter function f, integer k, and fixed integer B, is there a partition, $\mathcal{Z}$, of V into k or fewer subsets such that $f(\mathcal{Z}) \leq B$?

Examples of such functions are obtained by letting $\bullet = +$ and $\bullet = \max$. The latter results in the problem we call **PVD** (partitioning vertices to minimize maximum diameter). *PVD* receives particular attention in this thesis, we hence formally define it.

Problem *PVD*: Partitioning Vertices to Minimize Maximum Diameter: Given $G = (V, E)$, integer k, and fixed integer B, is there a partition, $\mathcal{Z}$, of V into k or fewer subsets such that $f(\mathcal{Z}) = \max_i\{diam(Z_i) : Z_i \in \mathcal{Z}\} \leq B$?

A special case of PVD is that of *partitioning vertices into cliques*, PVC, which we obtain by setting B to 1.

The statement of the second problem studied in this thesis now follows.

> **Problem** CXE **: Finding Cluster with Maximum Number of Edges**: Given $G = (V, E)$ and integers k and B, does G contain a connected induced subgraph $< W >$ such that the number of vertices in W is less than or equal to k and the number of edges in $< W >$ is greater than or equal to B?

A special case of CXE is the problem of finding the maximum clique which we obtain by setting $B = k(k-1)/2$.

Since the above problems are *NP*-complete on general graphs, we will restrict our attention to special graph families and obtain some complexity results and algorithms.

1.5 Previous Results and Contribution

All graph families mentioned in this section are defined in Appendix B.

Problem $PVMF$ may be easily shown to be *NP*-complete on general graphs; partitioning into cliques, a special case of $PVMF$ in *NP*-complete on general graphs.

We first survey results on various special cases of $PVMF$, then talk about our contribution to that problem and where it fits among previous results.

Berman et al. [8] prove that the special case PVC (Partitioning Vertices into Cliques) is *NP*-complete on planar graphs. Grötschel, Lovász and Schrijver [45] show that it has a polynomial time solution on perfect graphs. Special algorithms to solve PVC on chordal graphs, permutation graphs, comparability and cocomparability graphs are found in [34], [14], and [39]. Sequential and parallel efficient algorithms to solve it on interval graphs exist; see [46] for a sequential algorithm, and for parallel algorithms see [60], [10], and [62].

PVD (Partitioning Vertices to Minimize Maximum Diameter) is trivial if B is equal to the diameter of the graph. Thus, the problem is easy for cographs, for instance, since they are perfect graphs and have diameter less than or equal to two [22].

An algorithm to solve PVD on trees is presented in [31] by Farley, Hedetniemi, and Proskurowski. The algorithm starts from the bottom of the tree, and moves up. Edges are removed from the tree so the remaining forest consists of trees with bounded diameter B. An edge (u, v) is removed, where u is the father of v, if the diameter of the subtree rooted at u is greater than B and the subtree rooted at v has the largest height among all subtrees rooted at any of u's sons. The algorithm has a linear time complexity.

We now discuss our contribution with regards to $PVMF$. The solvability of PVD on trees raises the question of its status on chordal and bipartite graphs, both being generalizations of trees. In this thesis we show that PVD is *NP*-complete on chordal graphs, even when $B = 2$ and for split and undirected path graphs.

It should be noted that for $B = 1$ and $B > 2$, PVD on split graphs is solvable or trivial; when $B = 1$ it is solvable because split graphs are perfect, and when $B > 2$ it is trivial because they have diameter less than or equal to three.

On the positive side we present a linear time algorithm for PVD on interval graphs, which are a proper subset of chordal graphs.

For the general problem, $PVMF$, we present an efficient parallel algorithm on interval graphs.

We also show that PVD is *NP*-complete on bipartite graphs for all fixed values of B greater than one. However; we show that it has a linear time solution on bipartite permutation graphs.

As well we show how to solve $PVMF$ efficiently in parallel for biconvex graphs. To arrive at that result we first show that there is a new biconvex ordering for vertices of biconvex graphs, which we call *a biconvex straight ordering*, this ordering may be obtained efficiently in parallel.

Figure 1.2 summarizes the status of problems $PVMF$.

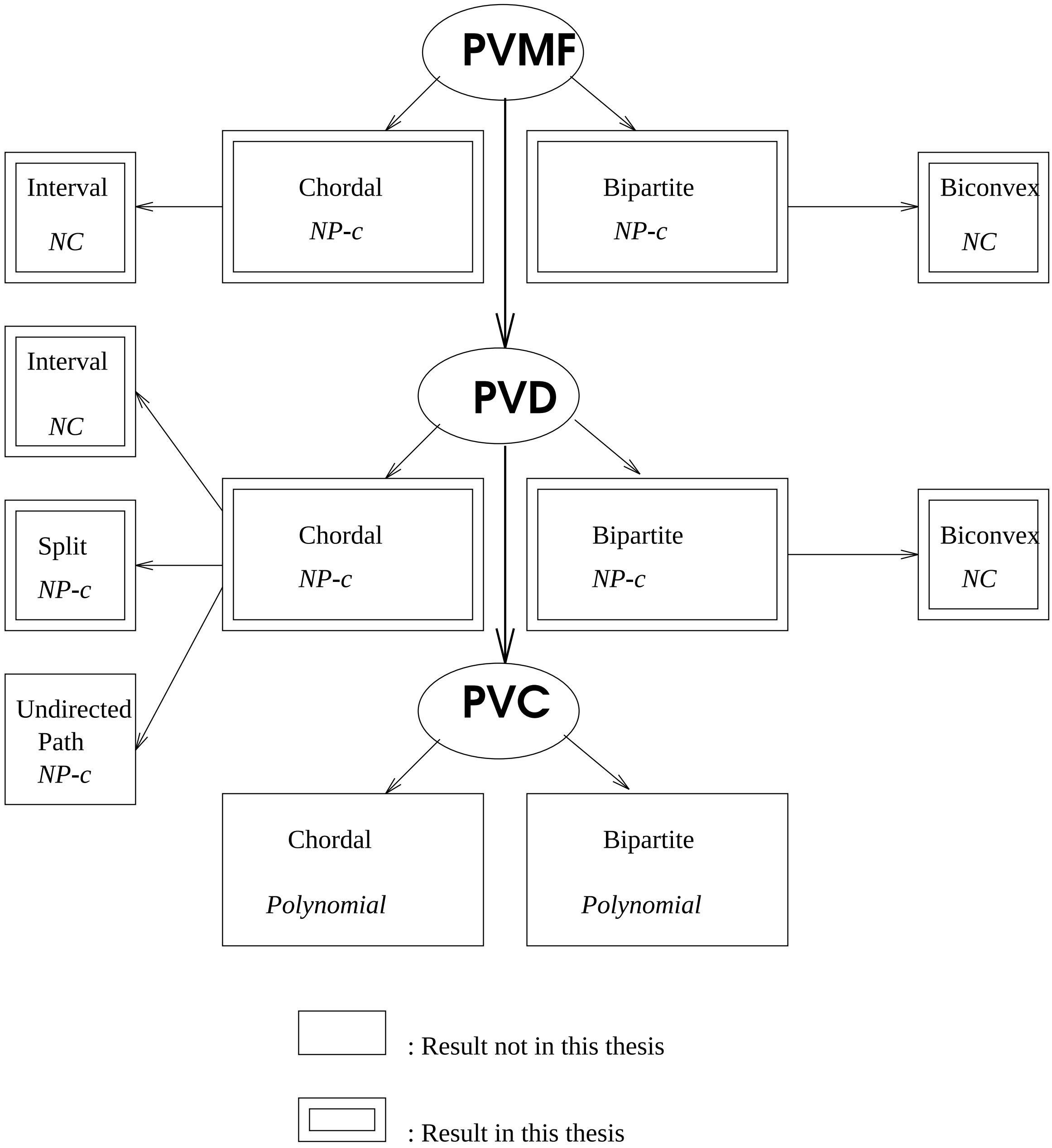

Figure 1.2: The status of problems $PVMF$

The second problem we consider is CXE. Results reported in the literature on this problem are mostly negative, i.e. *NP*-completeness results. It is easy to show that CXE is *NP*-complete on general graphs, as its special case, the maximum clique problem, is *NP*-complete on general graphs. It remains *NP*-complete when restricted to planar graphs, as

Keil and Brecht prove in [55], and when restricted to bipartite or chordal graphs as Corneil and Perl prove in [21].

It is not hard to see that CXE has a polynomial time solution for h-trees (as defined in Appendix B).

An algorithm to solve CXE on cographs is presented in [21]. The input to the algorithm is the cotree representation of the cograph, defined by Corneil, Lerchs, and Stewart in [22]. The algorithm is a dynamic program that starts from the bottom of the cotree, and works up. For each vertex the number of edges (*density*) of the best cluster on i vertices, for all possible i, is computed from the density of the clusters of its sons. The time complexity of the algorithm is $\mathcal{O}(k^2n)$, where k is the size of the desired cluster and n is the number of vertices in the graph.

Keil and Brecht [54] present an algorithm to solve CXE on partial h-trees for fixed h. Thus their algorithm solves the problem for a number of graph families including k-outerplanar [5], series parallel [72] and [74], Halin [11] , Δ- $\mathcal{Y}$ [27] and $\mathcal{Y}$-Δ [28], all of which have been proven to be partial h-trees for some fixed h.

The (edge-)weighted version of CXE is solved on trees in [64]. However it is shown to be *NP*-complete on cographs and split graphs even if the weights are binary [21].

Approximation algorithms for a higher set of weighted functions on trees is presented in [70] together with some *NP*-completeness results.

We present an efficient parallel algorithm to solve CXE on a family of unit interval graphs. Due to the scarcity of positive results on CXE, we hope our algorithm will shed light on the structures for which there is a polynomial time solution for CXE.

1.6 Organization of Thesis

This thesis is organized as follows. Chapter 2 introduces some *NP*-completeness results. The following four chapters aim at developing algorithms for the problem $PVMF$. In chapter three a parallel and a sequential algorithm for interval graphs are presented. Chapters four and five contain material that is necessary to design an algorithm for biconvex graphs. In chapter four we provide a new characterization of biconvex graphs, and make use of it in proving certain properties about distances and diameters in these graphs. In chapter five we use the results in chapter four to prove the existence of an optimal partition of a certain structure. Chapter six presents parallel and sequential algorithms for biconvex and bipartite permutation graphs respectively. Chapter seven contains results on CXE, a parallel algorithm for CXE and a class of unit interval graphs for which that algorithm gives the best cluster. Finally, we draw some conclusions, and discuss future research. Graph theory terminology is in appendix A. Definitions of various graph classes are in appendix B.

Appendix C contains definitions of complexity classes, complete problems, and the parallel computation model we use in this thesis.

Chapter 2

Partitioning Vertices to Minimize Maximum Diameter: Complexity

2.1 Introduction

As we mentioned in the introductory chapter, the problem of partitioning vertices of a graph into clusters with bounded diameter, PVD, has a linear time solution on trees [31]. Naturally, one wonders if the solvability of PVD on trees extends to chordal and bipartite graphs, both being generalizations of trees. In this chapter we show that such an extension is most probably not feasible. Particularly, we prove that PVD is *NP*-complete on bipartite and split graphs. The latter result extends to undirected path graphs.

We include the definitions of bipartite and split graphs in this chapter for the sake of completeness. A graph $G = (V, E)$ is bipartite if V can be partitioned into two sets X and Y, such that each induces an independent set, i.e. all edges in E have one endpoint in X and the other in Y. A graph $G = (V, E)$ is split if V can be partitioned into two sets Y_1 and Y_2, such that Y_1 induces an independent set and Y_2 induces a clique, i.e. no edge in E has both endpoints in Y_1, and all vertex pairs in Y_2 are adjacent.

Recall the statement of PVD.

> **Problem** PVD **: Partitioning Vertices to Minimize Maximum Diameter:** Given $G = (V, E)$, integer k, and fixed integer B, is there a partition, $\mathcal{Z}$, of V into k or fewer sets, such that $diam(Z_i) \leq B$, for all Z_i in $\mathcal{Z}$?

Also recall that PVD is one problem in a class of partitioning problems that we grouped under the title $PVMF$, which stands for: partitioning vertices to minimize monotone function.

A special case of PVD, is that of partitioning vertices into cliques, PVC, attainable by setting B to one. For both bipartite and chordal graphs PVC has a polynomial time solution (see [40] and [34]). In fact PVC is polynomially solvable for perfect graphs, which include both bipartite and chordal graphs [45].

For the purposes of our proofs we give definitions of the following *NP*-complete problems [33].

> $X3C$ **(Exact 3 Cover):** Given a set X s.t. $|X| = 3q$, and a collection $\mathcal{C}$ of subsets of X, s.t. $|C| = 3$ for all C in $\mathcal{C}$, does $\mathcal{C}$ have a subset $\mathcal{C}'$ such that every element in X occurs in exactly one member of $\mathcal{C}'$?

> PVC **(Partitioning Vertices into Cliques):** Given a graph $G = (V, E)$ and integer q, is there a partition, $\mathcal{Y}$, of V into q or fewer sets, such that each set Y_i in $\mathcal{Y}$ induces a clique?

Problem PVD is clearly in *NP*. Let $dist(u, v/W)$ denote the distance between vertices u and v in the subgraph induced by $W \subseteq V$, and $diam(W)$ denote the diameter of the subgraph, $< W >$, induced by $W \subseteq V$.

In the next section we study PVD on bipartite graphs.

2.2 PVD is *NP*-complete on Bipartite Graphs

Before presenting the proof that PVD is *NP*-complete on bipartite graphs for all fixed values of $B > 1$, we make some observations about relations between this problem and other problems, and about special cases of this problem.

As noted in [31], PVD on trees is equivalent to a *minimum edge deletion problem* where it is required to delete the minimum number of edges to obtain subgraphs with bounded diameter, because to obtain k subtrees we have to delete exactly $k - 1$ edges.

Also noted in that reference, PVD on trees with $B = 2$ is equivalent to partitioning vertices of the tree into the minimum number of sets so that each set contains a vertex that dominates all other vertices in the set. In other words all trees of diameter two contain a dominating vertex and each dominating vertex in a tree together with its neighbors induce a subtree of diameter two. This equivalency does not extend to bipartite graphs in general (for example a set isomorphic to $K_{2,2}$ has diameter two but does not contain a vertex that dominates all other vertices in the set). However, on bipartite graphs in general, PVD with $B = 2$, reduces to the problem of partitioning the vertices of a graph into the minimum number of complete bipartite graphs, which has not been proven *NP*-complete prior to the publication of this thesis. An interesting relation, nevertheless, exists between PVD on bipartite graphs and another domination problem. PVD with $B = 3$ is equivalent to finding a minimum cardinality dominating set whose members are either vertices or edges (a mixed dominating set).

A maximum matching for a bipartite graph is a solution of PVD with $B = 1$, where k (the cardinality of the optimal partition) is the number of edges in a maximum matching plus the number of unsaturated vertices (i.e. those vertices not incident with an edge in the maximum matching). Therefore, if the size of the maximum matching is k_1, then $k = |V| - k_1$. In the opposite direction a solution for the non-decision version of PVD with $B = 1$ (i.e. with a list of the sets in the optimal partition), offers a solution for the maximum matching problem.

To prove that PVD is *NP*-complete for bipartite graphs we consider two separate cases, B is even, and B is odd and greater than one. To prove the result when B is even we reduce $X3C$ to PVD. To construct a bipartite graph from the instance of $X3C$ we start with a base graph for the case $B = 2$ and add a polynomial number of edge subdividing vertices for all other values of B. To prove the result when B is odd and greater than one, we reduce PVD on general graphs to PVD on bipartite graphs. As in the first case we start with a base graph and add a number of edge subdividing vertices depending on the value of B.

Theorem 2.1 *PVD is NP-complete on bipartite graphs when B is a fixed even integer.*

Proof : We reduce $X3C$ to PVD. Given an instance of $X3C$, a set X such that $|X| = 3q$ and a set $\mathcal{C}$ of subsets of X such that $|C_i| = 3$ for all C_i in $\mathcal{C}$, we construct the bipartite graph $G^b = (V, E)$ where $V = L_0 \bigcup L_1 \bigcup L_2 \bigcup L_3 \bigcup S_1 \bigcup S_2$. These sets satisfy: for every x_i in X there is a vertex v_{i0} in L_0, for every C_i in $\mathcal{C}$ there is a vertex v_{i1} in L_1, each of L_2 and L_3 consists of one vertex, w and r respectively, and each of S_1 and S_2 contains edge-subdividing vertices.

Let $E^i = \{(r, w)\} \bigcup \{(w, v_{i1}), \forall i, 1 \leq i \leq |\mathcal{C}|\} \bigcup \{(v_{i1}, v_{j0})$ if $x_j \in C_i\}$.

E is obtained from E^i by edge subdivision as follows. Vertices of S_2 subdivide the edge (r, w) and $|S_2| = (B/2) - 1$. Vertices of S_1 subdivide edges between L_1 and L_0 such that for each edge there are $(B/2) - 1$ subdividing vertices, hence $|S_1| = ((B/2) - 1)(3|\mathcal{C}|)$. Figure 2.1 shows an example of the above reduction for the instance $X = \{x_1, ...x_9\}$, $\mathcal{C} = \{\{x_1, x_2, x_3\}, \{x_3, x_4, x_5\}, \{x_4, x_5, x_6\}, \{x_5, x_7, x_8\}, \{x_7, x_8, x_9\}\}$, and $B = 4$. It is clear that this transformation may be done in polynomial time.

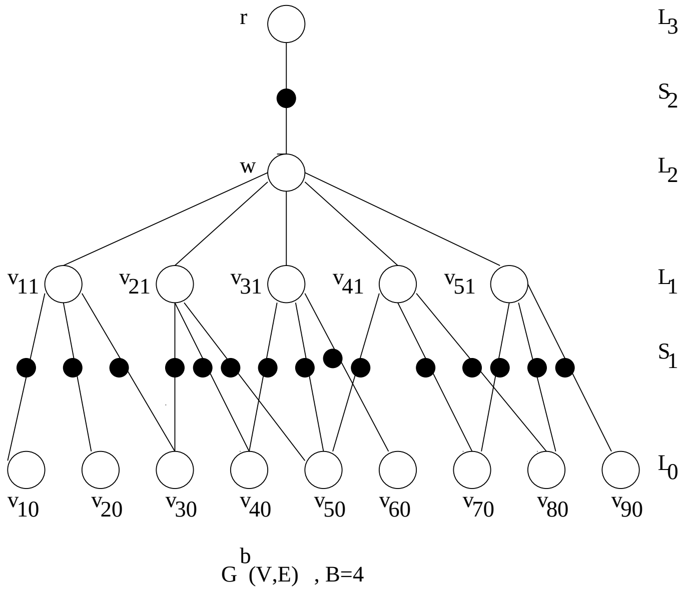

Figure 2.1: Reduction for the case B is even

We now prove that there is an exact three cover of size q for $X3C$ if and only if there is an optimal $q+1$ partition for G^b.

($\rightarrow$): Let the exact three cover be $\mathcal{C}' = \{C_1, ...C_q\}$ then $\mathcal{Z} = \{Z_1, \ldots, Z_{q+1}\}$ is a partition for G^b where $Z_j = \{v_{i0} : x_i \in C_j\} \bigcup \{v_{j1}\} \bigcup \{s : s \text{ subdivides an edge incident with } v_{j1}\}$, for all j, $1 \leq j \leq q$. $Z_{q+1} = V - \bigcup_{1 \leq j \leq q} Z_j$. Observe that Z_{q+1} contains no L_0 vertices. The following proves that $\mathcal{Z}$ is a valid partition:

First we show that $diam(Z_{q+1}) \leq B$. We note the following. For all s_i and s_j that belong to $S_1 \bigcap Z_{q+1}$, $dist(s_i, s_j/Z_{q+1}) \leq (B/2) - 1 + 2 + (B/2) - 1 = B$ (see figure 2.2 a). For every vertex v in Z_{q+1}, $dist(r, v/Z_{q+1})$ (and consequently $dist(w, v/Z_{q+1})$ and $dist(s, v/Z_{q+1})$ where $s \in S_2$) is less than or equal to B (see figure 2.2 a). For all v_{j1} and v_{i1} in Z_{q+1}, $dist(v_{i1}, v_{j1}/Z_{q+1}) = 2$. For every v_{j1} and s_i, $s_i \in S_1$, $dist(v_{j1}, s_i/Z_{q+1}) \leq 2 + (B/2) - 1 \leq B$ (see figure 2.2 a). Therefore $diam(Z_{q+1})$ is less than or equal to B.

Next consider $Z_1, \ldots, Z_l, \ldots, Z_q$ and observe the following:

$diam(Z_l) = dist(v_{i0}, v_{j0}/Z_l) = B$, for any v_{i0}, v_{j0} in Z_l (figure 2.2 b). Thus $\mathcal{Z}$ is a valid partition.

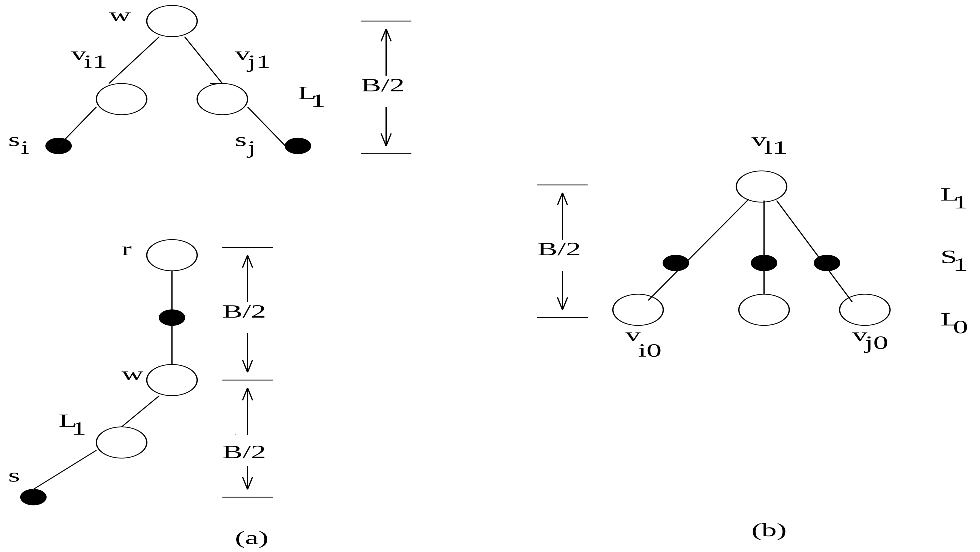

Figure 2.2: (a) Different Paths in Z_{q+1}
(b) Paths in Z_l, where $l \leq q$

($\leftarrow$): Any valid partition $\mathcal{Z}$ has to satisfy the following.

(a) For all v_{i0} and v_{j0} in any set Z_a in $\mathcal{Z}$, v_{i0} and v_{j0} must each have a path to a common vertex in L_1 through vertices of S_1. Otherwise $dist(v_{i0}, v_{j0}/Z_a)$ would be greater than B (figure 2.3 a).

(b) Since $dist(r, v_{i0}) > B$ for all v_{i0}, therefore r has to be in a set that does not contain any vertices from L_0. Let r belong to Z_{q+1}.

(c) Therefore L_0 has to be covered by q or fewer sets. Assume L_0 can be covered by fewer than q sets. Then there is a set Z_b in $\mathcal{Z}$ such that $|Z_b \bigcap L_0| > 3$. By (a) every pair in $Z_b \bigcap L_0$ must have a common neighbor in $Z_b \bigcap L_1$. Since $|Z_b \bigcap L_0| > 3$ and each v_{i1} in Z_b is adjacent to at most three vertices in $Z_b \bigcap L_0$, then there are vertices v_{i0} and v_{l1} in Z_b such that the path between v_{i0} and v_{l1} in Z_b passes through vertices not in S_1, and v_{l1} is connected to some other vertex v_{j0}, in Z_b, through vertices in S_1 only.

If $B = 2$ then $dist(v_{l1}, v_{i0}/Z_b) \geq 3$. Therefore in this case no set can contain more than three vertices from L_0. Otherwise $B > 2$. Any shortest path between v_{i0} and a vertex s_1 that subdivides the edge (v_{j0}, v_{l1}) and is as close as possible to L_0, either includes v_{j0} or includes w (see figure 2.3 b). Consider such a vertex s_1. The shortest path from s_1 to v_{i0} through w is of length $B/2 - 1 + 2 + B/2 > B$, and the shortest path from s_1 to v_{i0} through v_{j0} is of length $1 + B/2 + B/2 > B$. Therefore it must be true that $|Z_i \bigcap L_0| = 3$ and $Z_i \bigcap L_1 \neq \phi$ for all Z_i.

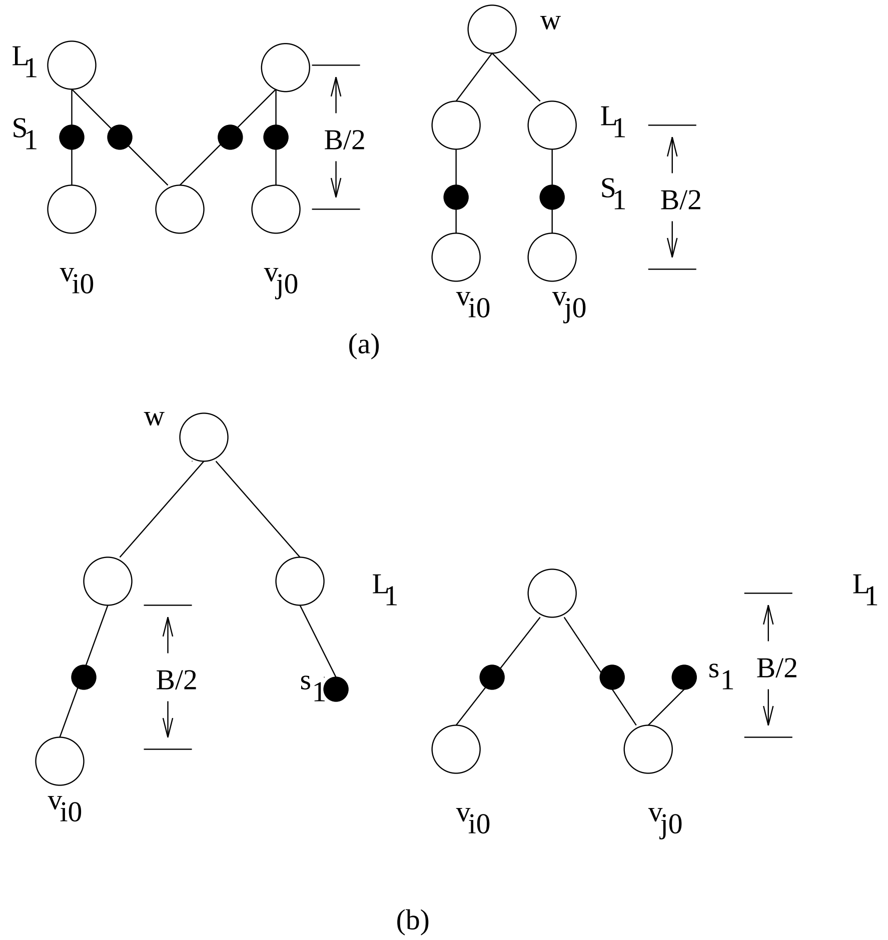

Figure 2.3: (a) Paths of length greater than B between v_{i0} and v_{j0}
(b) Paths of length greater than B between s_1 and v_{i0}

(d) Suppose $|Z_b \bigcap L_0| = 3$ but $|Z_b \bigcap L_1| > 1$ for some Z_b. It is safe to assume that no two vertices in L_1 are connected to the same set of vertices in L_0 (by paths through vertices of S_1). Then by (a) Z_b is isomorphic to, or contains an induced subgraph isomorphic to, one of the graphs in figure 2.4. But as in (c): if $B = 2$ then there are vertices v_{l1} and v_{i0} of distance greater than two, and if $B > 2$ then we can find s_1 and v_{i0} such that $dist(s_1, v_{i0}) > B$. Therefore $|Z_b \bigcap L_1| = 1$ and each v_{i1} in Z_b is adjacent to exactly three vertices in $Z_b \bigcap L_0$.

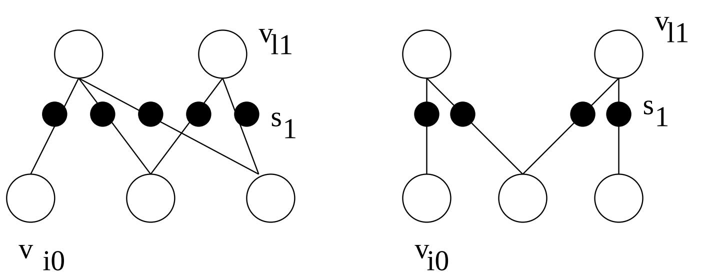

Figure 2.4: Illustration to show that Z_b must intersect L_1 in exactly one vertex

From the above we can see that each $Z_i, 1 \leq i \leq q$ corresponds to an exact three cover.■

In the next theorem we prove that PVD is *NP*-complete on bipartite graphs for all odd values of $B > 1$. This will complete the proof for all values of $B > 1$ on bipartite graphs.

Theorem 2.2 *PVD is NP-complete on bipartite graphs when B is a fixed odd integer greater than one.*

Proof : Given an instance of PVD on general graphs, $\{G = (V, E), B_G, q\}$, we construct an instance of PVD on bipartite graphs, $\{G^b = (V^b, E^b), B, q\}$, as follows. $V^b = L_0 \bigcup L_1 \bigcup L_2 \bigcup S$, where $L_0 = \{v_{i0} : \exists v_i \in V\}$, $L_1 = \{v_{j1} : \exists e_j \in E\}$, $L_2 = \{w_1, w_2, ...w_q\}$, and S is a set of edge-subdividing vertices. Let $E^i = \{(v_{i0}, v_{j1}) : e_j$ is incident with $v_i\} \bigcup \{(w_i, v_{j1}) : \forall i, j\}$.

E^b is obtained from E^i by adding the subdividing vertices S. Vertices of S subdivide the edges between L_1 and L_2 such that $B_G - 1$ vertices subdivide each edge. Therefore, $|S| = (B_G - 1)|E|q$.

Let S_i be the subset of S that subdivides edges incident with w_i.

Figure 2.5 shows an example of the above reduction for $B_G = 2$ and $q = 3$.

It is clear that this reduction may be done in polynomial time.

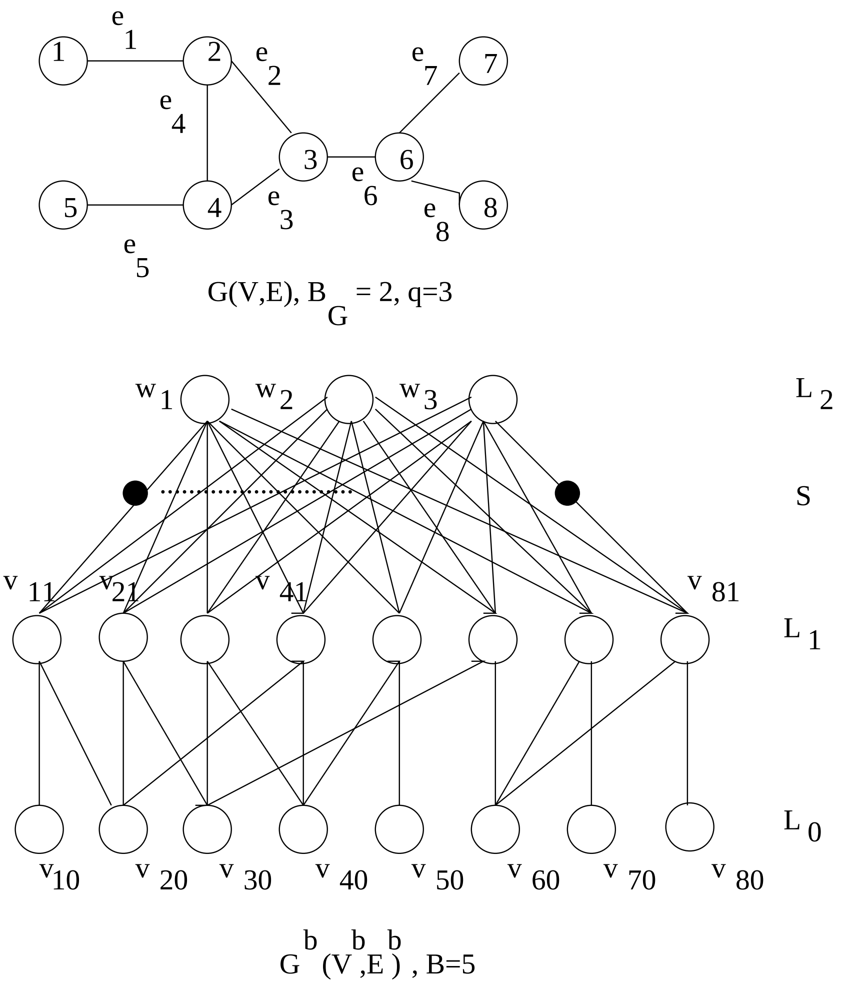

Figure 2.5: Reduction for the case B is odd

We now prove that there is a $q-$partition for G with diameters $\leq B_G$ if and only if there is a $q-$partition for G^b with diameters $\leq B = 2B_G + 1$.

($\rightarrow$): Let $\mathcal{Y}$ be a $q-$partition for G, $\mathcal{Y} = \{Y_1, Y_2, \ldots, Y_q\}$. For all Y_i, the edge e_j is in $< Y_i >$ if both endpoints of e_j are in Y_i. Then $\mathcal{Z} = \{Z_1, Z_2, \ldots, Z_q\}$ is a $q-$partition for G^b, where $Z_i = \{v_{j0} : v_j \in Y_i\} \bigcup \{v_{j1} : e_j \in < Y_i >\} \bigcup \{w_i\} \bigcup \{s : s \in S_i\}$. Each v_{j1} not in any set so far can be added to some set Z_a such that v_{j1} has a neighbor in Z_a. Each such v_{j1} is added to such a set Z_a with $|Z_a \bigcap L_0| = 1$, if possible. Notice that each such set Z_a would be disconnected unless some vertex v_{j1} is added to it. Since every vertex in G corresponding to $Z_a \bigcap L_0$ will have at least one edge to the rest of the graph, there will be more vertices v_{j1} not in any set so far than there are sets Z_a with $|Z_a \bigcap L_0| = 1$. Now all sets Z_i satisfy $|Z_i| > 1$ and $|Z_i \bigcap L_1| \geq 1$. Any v_{j1} not in any set so far can be added to any set it has neighbors in. The following properties satisfied by $\mathcal{Z}$ prove that it is a valid partition: $\forall Z_a$ in $\mathcal{Z}$

$\forall v_{i0}, v_{j0}$ in Z_a, $dist(v_{i0}, v_{j0}) \leq 2B_G < B$.

$\forall v_{i0}, v_{j1}$ in Z_a, $dist(v_{i0}, v_{j1}) = 1$ if e_j is incident with v_i, otherwise $dist(v_{i0}, v_{j1}) \leq 2B_G + 1 = B$ (figure 2.6 a).

$\forall v_{i0}, w_a$ in Z_a, $dist(v_{i0}, w_a) = B_G + 1 < B$ (figure 2.6 a).

$\forall v_{i0}, s$ in Z_a, $dist(v_{i0}, s) \leq 2B_G < B$ (figure 2.6 a).

$\forall v_{i1}, v_{j1}$ in Z_a, $dist(v_{i1}, v_{j1}) \leq 2B_G < B$ (figure 2.6 b).

$\forall v_{i1}, w_a$ in Z_a, $dist(v_{i1}, w_a) = B_G < B$ (figure 2.6 b).

$\forall v_{i1}, s$ in Z_a, $dist(v_{i1}, s) \leq 2B_G - 1 < B$ (figure 2.6 b).

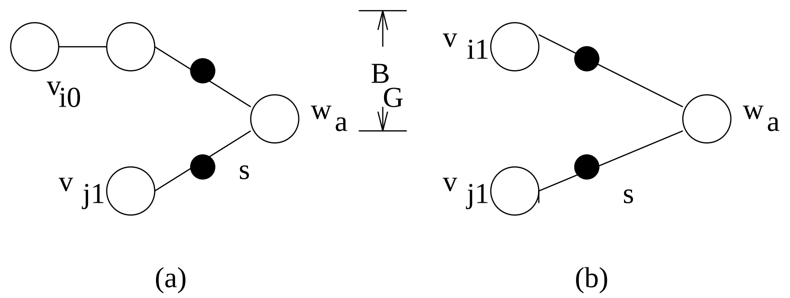

Figure 2.6 **(a) Paths in Z_a from v_{i0} to other vertices**
(b) Paths in Z_a from v_{i1} to other vertices

(←): Any valid partition has to satisfy the following property: for all Z_a in $\mathcal{Z}$, for all v_{i0}, v_{j0} in Z_a, $dist(v_{i0}, v_{j0}) \leq B - 1$ (since B is odd). Since any path between v_{i0} and v_{j0} that passes through L_2 is of length at least $2B_G + 2 > B$ (figure 2.7), therefore all paths between v_{i0} and v_{j0} in Z_a must be through vertices in $Z_a \bigcap (L_1 \bigcup L_0)$. Since $dist(v_i, v_j) = (1/2)dist(v_{i0}, v_{j0})$, therefore $Z_a \bigcap L_0$ induces a set of diameter $\leq B_G$ in G. ■

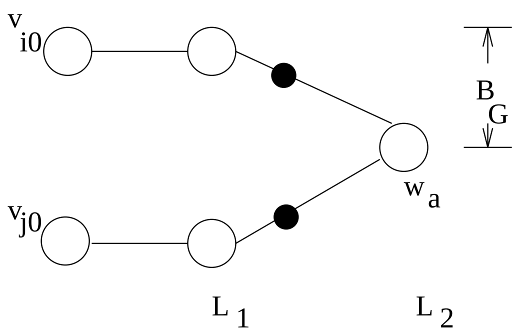

Figure 2.7: A path between v_{i0} and v_{j0} through L_2

Corollary 2.1 *PVD is NP-complete on bipartite graphs.*

Corollary 2.2 *PVD is NP-complete on comparability graphs.*

Corollary 2.3 *PVD is NP-complete on perfect graphs.*

2.3 *PVD* is *NP*-complete on Split Graphs

In this section we prove that *PVD* with $B = 2$ is *NP*-complete on split graphs. We published this result in [1], and it was obtained independently by the authors of [26]. The result extends to undirected path graphs as well since the graph constructed in the proof is both a split and an undirected path graph as first observed in [26].

We should note that PVD with $B=2$ on chordal graphs is equivalent to the *minimum dominating set* problem. A solution of PVD with $B=2$ on chordal graphs offers a solution for the minimum dominating set problem. In the opposite direction a solution of the minimum dominating set problem may be easily extended to one for PVD with $B=2$. Thus the results in this section may follow from the *NP*-completeness results in the literature for the minimum dominating set problem (see [21] and [9] for results on split graphs, and see [12] for the result on undirected path graphs).

Let $G=(V,E)$ be a split graph, where $V=Y_1 \bigcup Y_2$, Y_1 induces an independent set and Y_2 induces a clique.

It is interesting to note that PVC is easy on split graphs since a clique partition may be obtained by taking the largest clique (Y_2 with zero or one vertex from Y_1) as one set and every other vertex in Y_1 as a single-vertex set. Note also that split graphs have diameters less than or equal to three. In summary, for split graphs, PVD with $B=1$ is easy, and PVD with $B \geq 3$ is trivial. Therefore proving PVD *NP*-complete on split graphs when $B=2$ covers all values of B for this family of graphs.

We reduce PVC on general graphs to PVD on split graphs. The reduction is a simple one with a vertex in the split graph for every vertex or edge in the general graph. Vertices corresponding to edges form a clique in the split graph.

Theorem 2.3 (Also see [21], [9], [1]) *PVD is NP-complete on split graphs.*

Proof: We reduce PVC to PVD. Given an instance of PVC, $G=(V,E)$, we construct the split graph $G_s=(Y_1,Y_2,\mathcal{E})$ such that: for every vertex v_i in V there is a vertex y_i^1 in Y_1, and for every edge e_i in E there is a vertex y_i^2 in Y_2. The edge set is $\mathcal{E}=\{(y_i^2,y_j^2)\forall i,j\} \bigcup \{(y_i^1,y_j^2):$ *if v_i is incident with e_j*$\}$.

The following is true about G_s: $dist(y_i^2,y_j^2)=1$ for all i,j, $dist(y_i^1,y_j^2)=1$ if v_i is incident with e_j, otherwise it is equal to 2, and $dist(y_i^1,y_j^1)=2$ if v_i is adjacent to v_j, otherwise it is equal to three. Figure 2.8 shows an example of the above reduction.

It is clear that this reduction may be done in polynomial time.

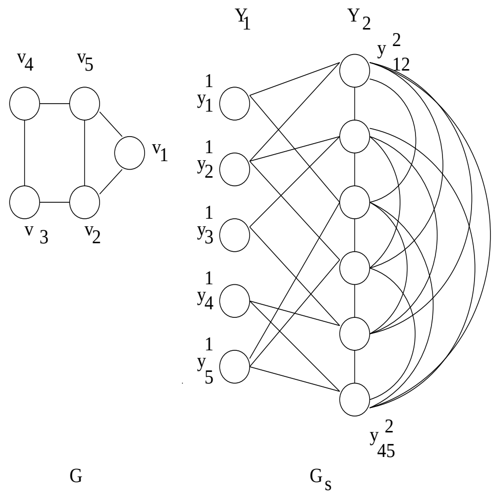

Figure 2.8: Reduction for the proof of theorem 2.3

We now prove that there is a q-clique cover for G if and only if there is a q-partition for G_s with $B = 2$.

($\rightarrow$): Let the clique cover be $\mathcal{C}= \{C_1, ...C_q\}$. The q-partition $\mathcal{Z} = \{Z_1, ...Z_q\}$ is such that: $Z_i = \{y^1_j : v_j \in C_i\} \bigcup \{y^2_j : e_j \in E_i\}$ for all $i = 1, 2, ...q$, where E_i is the set of edges in the subgraph, $< C_i >$, induced by C_i.

Since $\mathcal{C}$ covers all vertices of V, $\mathcal{Z}$, thus far, covers all vertices of Y_1. For every y^2_j not in $\mathcal{Z}$, y^2_j is adjacent to some vertex in $\mathcal{Z}$ (since the edge e_j has some endpoint in G), we add every such y^2_j to some set it has neighbors in.

Since every pair y^1_j and y^1_i in any set has a common neighbor y^2_o corresponding to $e_o = (v_j, v_i)$ therefore $diam(Z_i) \leq 2$ for all $i = 1, ..q$.

($\leftarrow$): Let the q-partition for G_s be $\mathcal{Z} = \{Z_1, ...Z_q\}$. Let $Z_1, ...Z_r$ be sets with diameter less than or equal to one. Let $Z_{r+1}, ...Z_q$ be sets with diameter equal to two. Then each of $Z_1, ...Z_r$ is one of the following.

(i) One vertex from Y_1, let these sets be $Z_1, ...Z_{r_1}$.

(ii) A clique from Y_2, let this set be Z_{r_1+1}.

(iii) A vertex from Y_1, y^1_o, and a clique from Y_2 such that y^1_o is adjacent to all vertices in the clique. Let these sets be $Z_{r_1+2}, ...Z_r$.

Also, each of $Z_{r+1}, ...Z_q$ must satisfy the following, $|Z_i| > 1$ otherwise it has diameter less than 2, it has to include at least one vertex from Y_2 and one vertex from Y_1 otherwise it is either disconnected or has diameter one, and every y^1_i and y^1_j in any Z_{r+i_1} must have a common neighbor in $Z_{r+i_1} \bigcap Y_2$ (i.e. an edge joining v_i and v_j in G) otherwise $dist(y^1_i, y^1_j) = 3$.

Therefore vertices of Y_1 are covered by $\mathcal{X} \subseteq \mathcal{Z}$, $\mathcal{X} = \{Z_1, ...Z_{r_1}, Z_{r_1+2}, ...Z_r, Z_{r+1}, ...Z_q\}$, and vertices of G corresponding to $Z_i \bigcap Y_1$, for all Z_i in $\mathcal{X}$, induce a clique in G. Thus G is covered by $\leq q$ cliques. ∎

Corollary 2.4 ([26], [12]) *PVD with $B = 2$ is NP-complete on undirected path graphs.*

Proof : The graph constructed in the proof of the previous theorem may be shown to be an undirected path graph. For instance, the graph G_s in figure 2.8 is the undirected path graph for the paths $\mathcal{P}$ of the tree T shown in figure 2.9. ∎

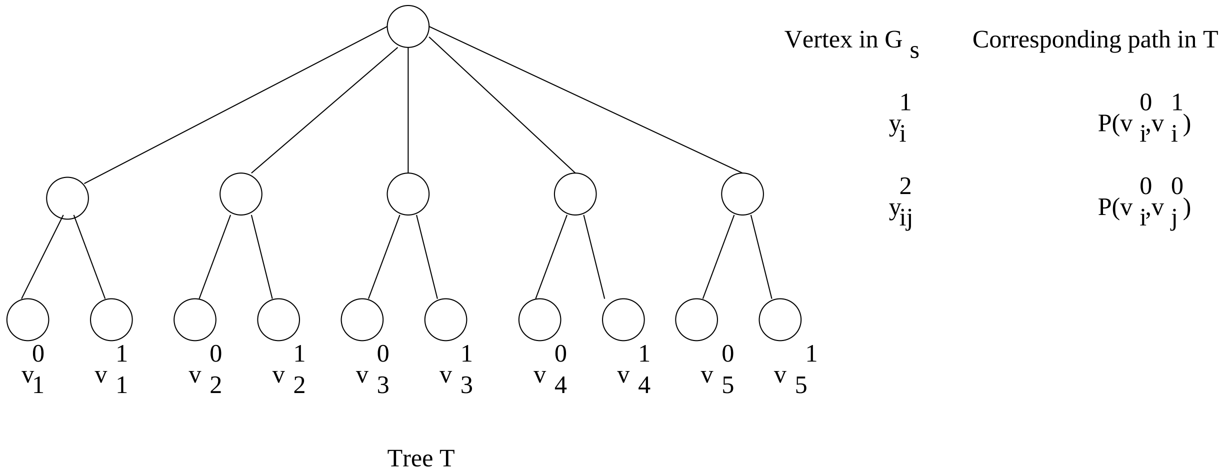

Figure 2.9: Paths in T for the undirected path graph G_s

Corollary 2.5 *PVD is NP-complete on chordal graphs.*

Chapter 3

Partitioning Interval Graphs

3.1 Introduction

Having established that PVD is hard on chordal graphs in general, we now proceed to prove that it has a linear time solution on interval graphs, and that the general problem $PVMF$ can be solved efficiently in parallel on interval graphs.

Interval graphs have wide applicability. We mentioned in the introductory chapter of this thesis that they arise in applications such as scheduling, VLSI, sociology, and archeology, (for examples of such applications see [68] and [40]). In addition, they have an ordering of the vertices that makes them algorithmically appealing.

Interval graphs are the intersection graphs of intervals in the real line. They are also exactly graphs that are both chordal and cocomparability [40].

In this chapter we present an efficient parallel algorithm for $PVMF$ on interval graphs (this algorithm is also published in [1]). We also present a linear time sequential algorithm for PVD, a special case of $PVMF$, on the same family of graphs. Recall problem $PVMF$.

> **Problem $PVMF$: Partitioning Vertices to Minimize Monotone Function:** Given a graph $G = (V, E)$, a monotone diameter function f, and non negative integers k and B, is there a partition $\mathcal{Z}$ of V into k or fewer sets such that $f(\mathcal{Z}) \leq B$?

Monotone diameter functions are defined in the introductory chapter (definition 1.1).

Problem PVD (Partitioning Vertices to Minimize Maximum Diameter) is a special case of this problem obtained by setting $\bullet = \max$.

Another special case of $PVMF$ is partitioning vertices into cliques (PVC), obtained by setting $\bullet$ to max and B to one. There are efficient sequential and parallel algorithms to solve PVC on interval graphs (see [46] for a linear time sequential algorithm, and for parallel algorithms see [10], [62] and [60]).

This chapter is organized as follows. The next section introduces the necessary definitions, and some proofs for useful properties of interval graphs. Section 3.3 contains the parallel algorithm for $PVMF$ and the necessary theorems to establish its correctness and complexity. Finally, the last section contains the sequential algorithm for PVD.

3.2 Preliminaries and Definitions

Let $G = (V, E)$ be an interval graph.

For every vertex v_i in V let $I_i = [a_i, b_i]$ be the corresponding interval. We will refer to interval i and vertex i (corresponding to interval i) interchangeably. Given a graph G, its interval representation (if one exists) may be obtained in parallel in polylogarithmic time using a polynomial number of processors (see [56] and [66]).

Let *head* and *tail* of v_i be $H(v_i) = a_i$ and $L(v_i) = b_i$, respectively.

Figure 3.1 shows an interval graph and its interval representation.

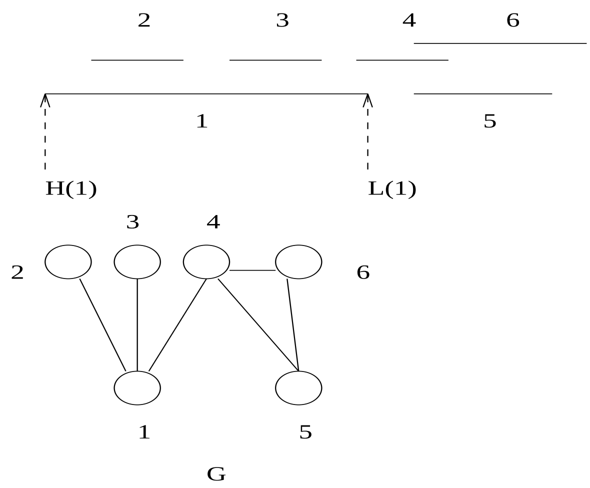

Figure 3.1: An interval graph G and its interval representation

Let vertices of G be ordered so that $v_i < v_j$ if $H(v_i) < H(v_j)$, or $H(v_i) = H(v_j)$ and $L(v_i) < L(v_j)$, otherwise ties are broken arbitrarily. This ordering may be obtained efficiently in parallel, using sorting [36], once we have the interval representation of G.

Let $\mathcal{P}(u, v/W)$ be the set of all shortest paths between u and v in the subgraph induced by W.

Let $P(u, v/W)$ be the set of vertices in an arbitrary element of $\mathcal{P}(u, v/W)$.

The *distance* between vertices u and v in the subgraph induced by $W \subseteq V$, $dist(u, v/W)$, is equal to $|P(u, v/W)| - 1$.

The *diameter* of the subgraph induced by $W \subseteq V$, $diam(W)$, is the maximum distance, through vertices in W, between any pair of vertices in W.

We will use the notation $P(u, v)$, $\mathcal{P}(u, v)$, and $dist(u, v)$ whenever we are considering the whole graph or when no confusion might arise.

$\mathcal{Z} = \{Z_1, ..., Z_k\}$ is an *optimal partition* if $f(\mathcal{Z}) \leq B$ and there is no other such partition for G of a smaller cardinality.

For every set Z_i in $\mathcal{Z}$, let $s(Z_i)$ be the first vertex in Z_i, $s(Z_i) = s_i = \min_{j'} v_{j'} \in Z_i$. Let $t(Z_i)$ be the last vertex in Z_i, $t(Z_i) = t_i = \max_{j'} v_{j'} \in Z_i$.

We define the relation $<_S$ on members of $\mathcal{Z}$ so that $Z_i <_S Z_j$ if $s_i < s_j$.

Let the *range* of Z_i be $R(Z_i)$ or $R_i = \{v : s_i \leq v \leq t_i\}$.

Z_i and Z_j are said to *overlap* if $R_i \bigcap R_j \neq \phi$.

The partition $\mathcal{Z}$ is called a *non-overlapping* partition if for all pairs Z_i and Z_j in $\mathcal{Z}$, Z_i and Z_j do not overlap.

Also define $min(Z_i) = min_i = \min_{L(v_j)} v_j \in Z_i$, i.e. min_i is the interval with the smallest value of tail among all intervals in Z_i. In case there is a tie let min_i be the vertex with the smallest index. Figure 3.2 illustrates the above definitions.

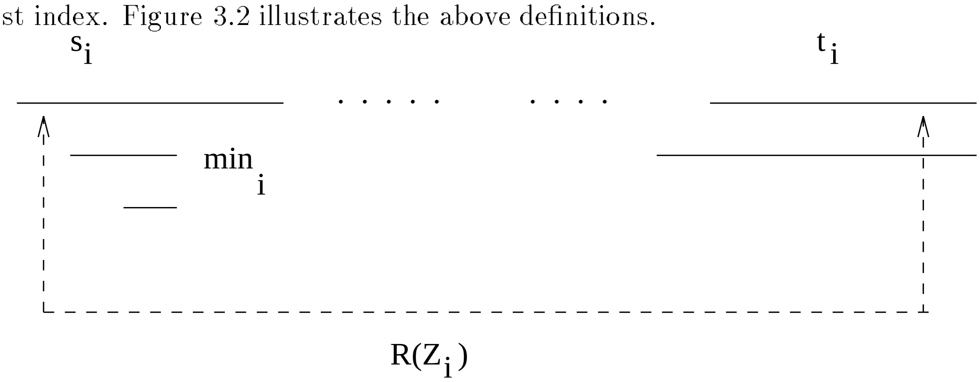

Figure 3.2: A set Z_i, its range, and various special intervals

We now prove some properties of interval graphs. First the following definition is necessary.

> **Definition 3.1:** Let $next(v) = \max_{L(u)} u$ *is adjacent to* v; i.e. $next(v)$ is the neighbor of v that extends the most to the right. In the case of a tie we choose the largest index vertex. If $next(v)$ does not exist we say it is equal to Λ (the null vertex).

Notice that $next(v)$ may be smaller or larger than v.

Proposition 3.1 (Also see [62]) *Given vertices u and w, $u < w$, let $P(u, w) = \{u = v_{i_0}, v_{i_1}, v_{i_2}, ...v_{i_p}, w = v_{i_{p+1}}\}$, be any path in $\mathcal{P}(u, w)$, then $v_{i_1} < v_{i_2} < ... < v_{i_p} < w$. Moreover if u is not adjacent to w, then there is a path $P(u, w)$ in $\mathcal{P}(u, w)$ such that $v_{i_j} = next(v_{i_{j-1}})$, for all j, $0 < j < p + 1$.*

Proof : Assume to the contrary that there are vertices v_{i_a} and $v_{i_{a+1}}$ in $P(u,w)$ such that $v_{i_a} > v_{i_{a+1}}$ and $v_{i_j} < v_{i_{j+1}}$ for all $j, 0 < j < a < p$. Therefore $H(v_{i_a}) \geq H(v_{i_{a+1}})$. If $L(v_{i_a}) \leq L(v_{i_{a+1}})$ then $v_{i_{a+1}}$ is adjacent to all neighbors of v_{i_a}, and v_{i_a} is not needed in $P(u,w)$, a contradiction. Therefore suppose $L(v_{i_a}) > L(v_{i_{a+1}})$.

If $a > 1$ then $v_{i_{a+1}}$ is adjacent to $v_{i_{a-1}}$, and v_{i_a} is not needed in $P(u,w)$ (a contradiction). If $a = 1$, then since $w > u$ therefore there is a vertex v_{i_b}, $b > a$, in $P(u,w)$ such that v_{i_b} is adjacent to u. Therefore there is a shorter path in this case as well. Therefore it must be true that $v_{i_1} < v_{i_2} < ... < v_{i_p} < w$.

Now suppose there is a smallest index b such that $v_{i_b} \neq next(v_{i_{b-1}})$, $v_{i_j} = next(v_{i_{j-1}})$, for all $j < b$. Notice that $L(next(v_{i_{b-1}})) \geq L(v_{i_b})$ (by definition 3.1 of $next(v_{i_{b-1}})$). For all $b \geq 2$, $v_{i_{b-1}} < v_{i_b}$ (by the argument in the above paragraph). Therefore if $2 \leq b < p+1$, or $b = 1$ and $v_{i_0} < v_{i_1} < v_{i_{p+1}}$ then $next(v_{i_{b-1}})$ may replace v_{i_b}. Else $b = 1$ and $v_{i_0} > v_{i_1}$, $p > 0$, and $v_{i_1} \neq next(v_{i_0})$. Since $w > u = v_{i0}$ and $v_{i_2} > v_{i_1}$ then $next(v_{i_0})$ is adjacent to v_{i_j}, for some $j \geq 2$, and may replace v_{i_1} in $P(u,w)$. ∎

Proposition 3.2 *For any set of (not necessarily connected) intervals* $\mathcal{I}$, $diam(\mathcal{I}) = dist(min(\mathcal{I}), t(\mathcal{I})/\mathcal{I})$.

Proof : If $\mathcal{I}$ is not connected then it must be true that $min(\mathcal{I})$ is not connected to $t(\mathcal{I})$. Otherwise every interval in $\mathcal{I}$ is adjacent to some interval in $P(min(\mathcal{I}), t(\mathcal{I}))$ (by the definition of $min(\mathcal{I})$ and of $t(\mathcal{I})$), and the set of intervals is connected. Therefore the proposition is true if $\mathcal{I}$ is not connected.

Now suppose $\mathcal{I}$ is connected, and there are vertices u^* and w^*, $u^* < w^*$, and $u^* \neq min(\mathcal{I})$ or $w^* \neq t(\mathcal{I})$, such that $dist(u^*, w^*) = diam(\mathcal{I})$ and $dist(min(\mathcal{I}), t(\mathcal{I})) < dist(u^*, w^*)$. Assume $u^* \neq min(\mathcal{I})$.

Due to the connectedness of $\mathcal{I}$, u^* and w^* are adjacent to vertices in $P(min(\mathcal{I}), t(\mathcal{I}))$. Since $dist(min(\mathcal{I}), t(\mathcal{I})/\mathcal{I}) < dist(u^*, w^*/\mathcal{I})$, then u^* is adjacent to the neighbor of $min(\mathcal{I})$ in $P(min(\mathcal{I}), t(\mathcal{I}))$ only, and w^* is adjacent to $t(\mathcal{I})$ only, in $P(min(\mathcal{I}), t(\mathcal{I}))$. But this implies $H(w^*) > H(t(\mathcal{I}))$, which contradicts the definition of $t(\mathcal{I})$. ∎

Proposition 3.3 *For any set of intervals* $\mathcal{I}$ *and any vertex* u *such that* $L(u) \geq L(min(\mathcal{I}))$, *and* u *is either in the range of* $\mathcal{I}$ *or is adjacent to* $min(\mathcal{I})$, $diam(\mathcal{I} \bigcup \{u\}) \leq diam(\mathcal{I})$.

Proof: Follows from proposition 3.2. ∎

Proposition 3.4 *For any set of intervals* $\mathcal{I}$, *let* $\mathcal{I}' = \mathcal{I} - \{t(\mathcal{I})\}$, *then* $diam(\mathcal{I}') \leq diam(\mathcal{I})$.

Proof : To prove this, note that, by proposition 3.2, $diam(\mathcal{I}') = dist(min(\mathcal{I}'), t(\mathcal{I}')) = dist(min(\mathcal{I}), t(\mathcal{I}'))$, and since $t(\mathcal{I}) > t(\mathcal{I}')$ then by proposition 3.1 $t(\mathcal{I}) \notin P(min(\mathcal{I}), t(\mathcal{I}'))$. Therefore removing $t(\mathcal{I})$ does not increase the diameter. ∎

3.3 Parallel Algorithm for $PVMF$

We now prove the main theorem that will enable us to find an efficient parallel algorithm. We show that there is an optimal partition with each set consisting of consecutive vertices in G (a non-overlapping partition). Then later, in the second theorem of this chapter, we will show how such a partition may be expressed as the union of two disjoint subpartitions of smaller cardinality, and how this directly gives an efficient parallel algorithm. We always deal with connected graphs.

Theorem 3.1 *Given an interval graph $G = (V, E)$, G has an optimal non-overlapping partition.*

Proof : Assume to the contrary that the claim is not true. Among all optimal partitions for G choose $\mathcal{Z}$ to be one with the minimum number of pairwise overlaps among its set pairs. We will choose a pair of sets, Z_i and Z_j, in $\mathcal{Z}$ that overlap. We will then define sets $Z_i^{'}$ and $Z_j^{'}$, of disjoint ranges, such that $diam(Z_i^{'}) \leq diam(Z_i)$, $diam(Z_j^{'}) \leq diam(Z_j)$, and $Z_i^{'} \bigcup Z_j^{'} = Z_i \bigcup Z_j$. Hence $\mathcal{Z}^{'} = \mathcal{Z} - \{Z_i, Z_j\} \bigcup \{Z_i^{'}, Z_j^{'}\}$ is an optimal partition (since $|\mathcal{Z}^{'}| = |\mathcal{Z}|$ and $f(\mathcal{Z}^{'}) \leq f(\mathcal{Z})$, recall the definition of a monotone diameter function). We then show that compared with $\mathcal{Z}$, $\mathcal{Z}^{'}$ has fewer pairwise overlaps, thus concluding our proof with a contradiction.

Let Z_i and Z_j be two sets of $\mathcal{Z}$ that overlap such that $L(min_i) \leq L(min_j)$. Let $dist(u, v)$ refer to $dist(u, v/Z_i \bigcup Z_j)$, and $P(u, v)$ refer to $P(u, v/Z_i \bigcup Z_j)$. There are two cases.

Case 1: $t(Z_i) > t(Z_j)$ (i.e. $Z_i <_S Z_j$ implies $R_j \subset R_i$ and $Z_i >_S Z_j$ implies $R_j \not\subset R_i$, as shown in figure 3.3 a).

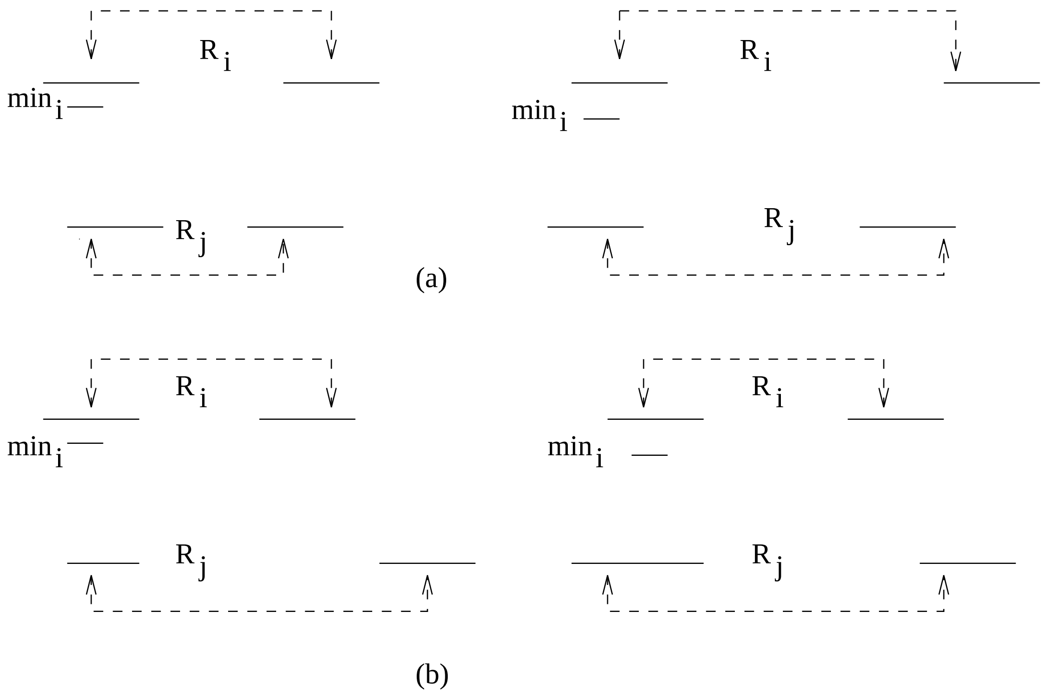

Figure 3.3: Illustration for theorem 3.1

In this case it follows directly from proposition 3.3 that $diam(Z_i \bigcup Z_j) \leq diam(Z_i)$. Let $\mathcal{Z}' = \mathcal{Z} - \{Z_i, Z_j\} \bigcup \{Z_i \bigcup Z_j\}$. Therefore $f(\mathcal{Z}') \leq f(\mathcal{Z})$ and $|\mathcal{Z}'| < |\mathcal{Z}|$ (a contradiction to the minimality of $\mathcal{Z}$).

Case 2: $t(Z_i) < t(Z_j)$ (i.e. $Z_i <_S Z_j$ implies $R_i \not\subset R_j$ and $Z_i >_S Z_j$ implies $R_i \subset R_j$, as shown in figure 3.3 b).

In this case let $Z_i' = \{v : v \in Z_i \bigcup Z_j, v \leq t(Z_i)\}$. $Z_j' = (Z_i \bigcup Z_j) - Z_i'$. According to proposition 3.3, $diam(Z_i') \leq diam(Z_i)$. Also $R_i' \bigcap R_j' = \phi$ (where $R_i' = R(Z_i')$ and $R_j' = R(Z_j')$), and $Z_j' \subseteq Z_j$.

If $diam(Z_j') \leq diam(Z_j)$ then let $\mathcal{Z}' = \mathcal{Z} - \{Z_i, Z_j\} \bigcup \{Z_i', Z_j'\}$. Therefore, $f(\mathcal{Z}') \leq f(\mathcal{Z})$ and $|\mathcal{Z}'| = |\mathcal{Z}|$. To prove that the number of overlaps in $\mathcal{Z}'$ is less than that in $\mathcal{Z}$, we observe the following. Z_i' and Z_j' do not overlap. The sets that overlap Z_i but not Z_j, now overlap Z_i' but not Z_j'. The sets that overlap Z_j but not Z_i, now overlap Z_i' or Z_j' but not both. We explain why the last statement is true. Suppose Z_c is a set that overlaps Z_j only. If $R_i \not\subset R_j$ then Z_c overlaps Z_j' only. If $R_i \subset R_j$ then one of the following is true: Z_c has no vertex larger than s_i and therefore does not overlap Z_j', or Z_c has no vertex smaller than t_i and therefore does not overlap Z_i'. Therefore the number of overlaps decreased by at least one. This contradicts the claim that we started with a partition that has the minimum number of pairwise overlaps.

Else $diam(Z_j^{'}) > diam(Z_j)$. In this case since initially $R_i \bigcap R_j \neq \phi$, therefore there is a set $VN \subseteq (Z_i^{'} \bigcap Z_j)$ such that $diam(VN \bigcup Z_j^{'}) \leq diam(Z_j)$. We prove the following claim first.

Claim 3.1.1 *There is a set $VN \subseteq (Z_i^{'} \bigcap Z_j)$ such that $diam(VN \bigcup Z_j^{'}) \leq diam(Z_j)$ and $|VN| = 1$.*

Proof of claim: We know that there is at least one such set. Furthermore there is one such set that is a subset of $P(min(Z_j^{'}), t(Z_j^{'})/Z_j)$, by proposition 3.2. Let VN be one such set. Assume that $|VN| > 1$, let $VN = \{vn_1, ...vn_p\}$, $p > 1$. Let the order of the vertices of VN be similar to their ordering in $P(min(Z_j^{'}), t(Z_j^{'})/Z_j)$. Since VN belongs to $Z_i^{'}$ therefore vn_p is smaller than $min(Z_j^{'})$. Thus vn_p is adjacent to both $min(Z_j^{'})$ and a vertex v, where v is the first vertex in $P(min(Z_j^{'}), t(Z_j^{'})/Z_j)$ not in VN. Therefore $VN = \{vn_p\}$ satisfies the claim. ∎

We next prove that we can add vn_p to $Z_j^{'}$ then separate the ranges again.

Claim 3.1.2 *Let $NB = \{v : vn_p < v < s(Z_j^{'})\}$, $Z_j^{''} = Z_j^{'} \bigcup \{vn_p\} \bigcup NB$ and $Z_i^{''} = Z_i^{'} - \{vn_p\} - NB$, then $diam(Z_j^{''}) \leq diam(Z_j)$ and $diam(Z_i^{''}) \leq diam(Z_i)$.*

Proof of claim: The proof that $diam(Z_i^{''})$ is less than or equal to $diam(Z_i)$ follows directly from proposition 3.4.

To prove that $diam(Z_j^{''})$ is less than or equal to $diam(Z_j)$, first note the following. $diam(Z_j^{'} \bigcup \{vn_p\}) \leq diam(Z_j)$ (by claim 3.1.1). Since $diam(Z_j^{''}) = dist(min(Z_j^{''}), t(Z_j^{''})/Z_j^{''})$ by proposition 3.2, then $diam(Z_j^{''}) > diam(Z_j^{'} \bigcup \{vn_p\})$ implies that $min(Z_j^{''}) \in NB$ (by proposition 3.3). Also vn_p is adjacent to $s(Z_j^{'})$ implies that all vertices of NB are adjacent to vn_p. By the previous claim and proposition 3.2 we know that vn_p belongs to $P(min(Z_j^{'}), t(Z_j^{'})/Z_j^{'} \bigcup \{vn_p\})$. Since, as well, $t(Z_j^{''}) = t(Z_j^{'})$ then it follows that $dist(min(Z_j^{''}), t(Z_j^{''})/Z_j^{''}) \leq dist(min(Z_j^{'}), t(Z_j^{'})/Z_j^{'} \bigcup \{vn_p\})$. Therefore $diam(Z_j^{''}) \leq diam(Z_j)$. ∎

Now we continue with the proof of the theorem.

Let $\mathcal{Z}^{'} \leftarrow \mathcal{Z} - \{Z_i, Z_j\} \bigcup \{Z_i^{''}, Z_j^{''}\}$. Then $|\mathcal{Z}^{'}| = |\mathcal{Z}|$, $f(\mathcal{Z}^{'}) \leq f(\mathcal{Z})$, and $\mathcal{Z}^{'}$ is optimal.

To prove that the number of overlaps decreased, note that $s(Z_j^{''}) = vn_p \geq s(Z_j)$. Therefore $R(Z_j^{''}) \subseteq R_j$, and any set that overlaps $Z_j^{''}$ overlaps Z_j. Also note that $t(Z_i^{''}) \leq t(Z_i)$. Now suppose some set Z_c overlaps both $Z_i^{''}$ and $Z_j^{''}$ but overlaps Z_j only, then it must be the case that $R_i \subset R_j$ and $s(Z_j^{''}) < s(Z_i)$. But this implies that $min_i \in Z_j^{''}$, and that for all $v \in Z_i^{''}$, $L(v) > L(min_i)$ and v is adjacent to min_i, with the consequence that $diam(Z_i^{''} \bigcup Z_j^{''}) \leq diam(Z_j^{''})$ (by proposition 3.3). This contradicts the minimality of $\mathcal{Z}$.

Therefore the number of overlaps decreased by at least one, which is a contradiction to our choice of $\mathcal{Z}$. ∎

The next theorem translates into a straightforward parallel algorithm. Let $\mathcal{Z}$ be an optimal partition for G, that satisfies theorem 3.1. In the next theorem we show that there are disjoint subpartitions, $\mathcal{Z}_1$ and $\mathcal{Z}_2$, of $\mathcal{Z}$ such that $f(\mathcal{Z}_1) \bullet f(\mathcal{Z}_2) \leq B$, $\mathcal{Z} = \mathcal{Z}_1 \bigcup \mathcal{Z}_2$, and each of $\mathcal{Z}_1$ and $\mathcal{Z}_2$ satisfies theorem 3.1. The effect of this theorem is to enable us find an optimal partition successively by building its smaller subpartitions, all in polylogarithmic time and using a polynomial number of processors.

Let $G = (V, E)$ be an interval graph, where $V = \{v_1, \ldots, v_n\}$. Let $\mathcal{Z}_i^j$ be an optimal partition, that satisfies theorem 3.1, for the subgraph induced by $\{v_i, \ldots, v_j\}$. Let $Q^*(i, j)$ be the cardinality of $\mathcal{Z}_i^j$.

Theorem 3.2 $Q^*(1, n) = \min_{\substack{1 \leq j \leq n \\ f(\mathcal{Z}_1^j) \bullet f(\mathcal{Z}_{j+1}^n) \leq B}} (Q^*(1, j) + Q^*(j+1, n))$.

Furthermore there is such an optimal partition $\mathcal{Z}$ where each of $Q^*(1, j)$ and $Q^*(j+1, n)$ is not larger than $\lceil (1/2) * Q^*(1, n) \rceil$.

Proof : We prove the claim by induction on $Q^*(1, n)$ and using the fact that there is an optimal non-overlapping partition.

The hypothesis is trivially true for $Q^*(1, n) = 1$. The hypothesis is also true for $Q^*(1, n) = 2$. Since any partition of cardinality two $\mathcal{Z} = \{Z_1, Z_2\}$, may be expressed as $\mathcal{Z} = \mathcal{Z}_1^{t(Z_1)} \bigcup \mathcal{Z}_{s(Z_2)}^n$, where $s(Z_2) = t(Z_1) + 1$. Assume it is true for $|\mathcal{Z}| < k$.

Let $\mathcal{Z} = \{Z_1, Z_2, \ldots, Z_k\}$ be an optimal partition for an interval graph G with n vertices, that satisfies theorem 3.1.

$f(\mathcal{Z}) = diam(Z_1) \bullet diam(Z_2) \bullet .... \bullet diam(Z_r) .... \bullet diam(Z_k)$
$= (diam(Z_1) \bullet diam(Z_2), ... \bullet diam(Z_r)) \bullet (diam(Z_{r+1}) ...... \bullet diam(Z_k))$ (since $\bullet$ is associative).

Therefore there are two subsets of $\mathcal{Z}$, $\mathcal{Z}_1$ and $\mathcal{Z}_2$, such that $\mathcal{Z}_1 \bigcup \mathcal{Z}_2 = \mathcal{Z}$, $\mathcal{Z}_1 = \{Z_1, Z_2, \ldots, Z_r\}$, $\mathcal{Z}_2 = \{Z_{r+1}, Z_{r+2}, \ldots, Z_k\}$, and $f(\mathcal{Z}_1) \bullet f(\mathcal{Z}_2) \leq B$.

Since $\mathcal{Z}_1 \neq \phi$, $\mathcal{Z}_2 \neq \phi$, and $|Z_i| \geq 1$ for all i, $1 \leq i \leq k$, therefore $1 \leq |\mathcal{Z}_1| \leq k - 1$, $1 \leq |\mathcal{Z}_2| \leq k - 1$, and $Q^*(1, n) = Q^*(1, t(Z_r)) + Q^*(s(Z_{r+1}), n)$ (where $s(Z_{r+1}) = t(Z_r) + 1$). Furthermore by the associativity of $\bullet$ there are such subpartitions that satisfy $|\mathcal{Z}_1|, |\mathcal{Z}_2| \leq \lceil 1/2 * |\mathcal{Z}| \rceil$. Since both terms on the right hand side are correctly computed by the induction hypothesis, then $Q^*(1, n) = k$ is correctly computed. ■

We now present an efficient parallel algorithm. To compute the cardinality, $Q^*(i, j)$, of the optimal partition, $\mathcal{Z}_i^j$, that covers a set of intervals $\mathcal{I} = \{i, i+1, ...j\}$, the algorithm tries all possible bipartitions of $\mathcal{I}$ into sets $\mathcal{I}_1 = \{i, i+1, ...l\}$ and $\mathcal{I}_2 = \{l+1, ...j\}$. If the cardinalities of the optimal partitions $\mathcal{Z}_i^l$ and $\mathcal{Z}_{l+1}^j$ have already been computed, for all l, then the result of theorem 3.2 can be used to compute $Q^*(i, j)$. The doubling technique (see for instance [36]) is used so that after each iteration of the main loop the cardinality of the subpartitions computed is doubled.

We use $F(i, j)$ to store $f(\mathcal{Z}_i^j)$. The variable $Q^*(i, j)$ is the cardinality of the optimal partition for the set of intervals $\{i, i+1, \ldots, j\}$. And $W[i, l, j] = Q^*(i, l) + Q^*(l+1, j) =$

$|\mathcal{Z}_i^l| + |\mathcal{Z}_{l+1}^j|$ (note that $\mathcal{Z}_i^l \bigcup \mathcal{Z}_{l+1}^j$ is not necessarily optimal).

Algorithm 3.1 Algorithm PVMF-INTERVAL(G, B, k)

Input: *Interval graph* $G = (V, E)$, where $|V| = n$, and *integer* B.
Output: *The cardinality* k *of the optimal partition,* $k = |\mathcal{Z}|$.

begin

1. **for all** i *and* j **do in parallel**
   $F(i,j) = diam(\{i, i+1, ...j\})$
   if $F(i,j) \leq B$ **then** $Q^*(i,j) = 1$ **else** $Q^*(i,j) = \infty$
   endfor
2. **repeat** $\lceil \log n \rceil$ *times*
   (a) **for all** i, l *and* $j, i \leq l < j$ **do in parallel**
   $W[i,l,j] = \infty$
   if $F(i,l) \bullet F(l+1,j) \leq B$ **then** $W[i,l,j] = Q^*(i,l) + Q^*(l+1,j)$
   endfor
   (b) **for all** i *and* j **do in parallel**
   $Q^*(i,j) = \min(\min_l W[i,l,j], Q^*(i,j))$
   if $Q^*(i,j) \neq W[i,l,j]$ *for some* l **then** $l^* = 0$
   else $l^* = \{l : Q^*(i,j) = W[i,l,j]\}$
   endif
   if $l^* \neq 0$ **then** $F(i,j) = F(i,l^*) \bullet F(l^*+1,j)$
   endfor
3. $k = Q^*(1,n)$.

end

We now prove that the algorithm needs polylogarithmic time and a polynomial number of processors on our parallel computation model *CREW-PRAM*.

Theorem 3.3 *Algorithm 3.1 computes the cardinality of the optimal partition in* $\mathcal{O}(\log^2 n)$ *time, using* $\mathcal{O}(n^3)$ *processors on a* CREW-PRAM.

Proof : Proof of correctness follows from theorem 3.2, and the fact that we try all possibilities.

To calculate the cost of the algorithm, it is shown in [62] that the distance between any two vertices in an interval graph can be computed in $\mathcal{O}(\log n)$ time using $\mathcal{O}(n)$ processors on a *CREW-PRAM*. We briefly show how to perform this computation. Let $next^0(v) = v$ and $next^i v = next(next^{i-1}(v))$ for $i > 0$. Calculate $next(v)$ for all v; this may be done in

$\mathcal{O}(\log n)$ time using $\mathcal{O}(n)$ processor as shown in [62]. Then find $dist(u, w)$ using the path $\{u, next(u), next(next(u)), \ldots, w\}$ as in proposition 3.1. For the last step use the doubling technique [36] so after the ith iteration each vertex v has a pointer to $next^{2^i}(v)$. Also $min(\mathcal{I})$ and $t(\mathcal{I})$ can be computed with the same cost using a well known technique of finding the minimum (or maximum) of n unsorted numbers [36]. Therefore according to the above and proposition 3.2, the diameter of a set of intervals can be computed in $\mathcal{O}(\log n)$ time using $\mathcal{O}(n)$ processors.

Now we calculate the cost of the other statements of the algorithm. Computing the value of $F(i, j)$, in step 1, has the same cost as computing the diameter of a set of intervals and the loop (step 1) needs n^2 processors, one for each pair (i, j). The main loop, in step 2, takes $\mathcal{O}(\log n)$ time . The first inner loop, step 2.a, needs n^3 processors, one for each triplet (i, l, j). $W[i, l, j]$ can be computed in constant time with one call to the operator $\bullet$. The second inner loop, step 2.b, needs n^2 processors one for each pair (i, j). The first statement in that loop calls for calculating the minimum of at most n unsorted numbers and therefore $\mathcal{O}(n)$ processors and $\mathcal{O}(\log n)$ time are sufficient [36]. Computing Q^* and F needs constant time. Therefore the algorithm requires $\mathcal{O}(\log^2 n)$ time and $\mathcal{O}(n^3)$ processors. ■

Remark: Problem $PVMF$ on interval graphs may be shown to be a *self reducible problem* (see [53] for definitions and for examples of such problems).

3.4 Sequential Algorithm for PVD

In this section we present an optimal sequential algorithm for PVD (Partitioning Vertices to Minimize Maximum Diameter). The algorithm has a lower complexity than the sequential version of algorithm 3.1. It uses Depth First Search (DFS).

We prove a useful property about optimal partitions satisfying theorem 3.1 for problem PVD. With very few modifications such a partition may be transformed into one with sets having nearly equal diameters.

The special case when $B = 1$, i.e. Partitioning Vertices into Cliques, is also studied in the literature. For example see [10], [62] and [60] where parallel algorithms to solve PVC are presented. In [46] an optimal sequential algorithm is presented for the same problem on interval graphs. It runs in linear time ($\mathcal{O}(n)$ time) if the intervals are already sorted, and in $\mathcal{O}(n \log n)$ otherwise. That algorithm makes use of the fact that interval graphs are perfect graphs, and hence the size of the optimal clique partition is equal to the size of the maximum independent set [40]. A maximum independent set is first sought. It is then extended into a clique cover by adding to each vertex in the maximum independent set some of its neighbors (in such a way that every vertex is in some set).

We will assume that the intervals are already sorted.

Let $G = (V, E)$ be an interval graph, and let $\mathcal{Z}$ be an optimal non-overlapping partition for G. Define the following.

> **Definition 3.2:** A set Z_i is said to be *maximal* if for every set U of consecutive vertices that are smaller than every vertex in Z_i, i.e. for all $u \in U$ $u < s_i$, $diam(Z_i \bigcup U) > B$.

The next proposition is a direct consequence of the above definition.

Proposition 3.5 *Given a vertex v, the maximal set Z_i that has $t(Z_i) = v$ is* unique.

We use the above proposition to show that there is a unique optimal partition that consists of maximal sets.

Theorem 3.4 (Near Equal Diameters Partition) *G has a unique optimal non-overlapping partition, where each set has diameter less than or equal to B, such that each set is maximal, and all sets, except possibly for the first, have diameter B or $B - 1$.*

Proof : Let $\mathcal{Z}$ be an optimal partition for G that satisfies theorem 3.1. Choose $\mathcal{Z}$ so that it has the largest number a of consecutive sets $Z_k, Z_{k-1}, \ldots, Z_{k-a+1}$ that satisfy this theorem.

Therefore $diam(Z_{k-a}) < B$, $a < k - 1$.

Let x^* be the smallest index vertex such that $diam(Z_{k-a} \bigcup U) \leq B$ where $U = \{x : x^* \leq x < s_{k-a}\}$. Since $\mathcal{Z}$ is optimal and non-overlapping therefore $U \subset Z_{k-a-1}$. Let $Z_{k-a} = Z_{k-a} \bigcup U$ and $Z_{k-a-1} = Z_{k-a-1} - U$. Since U consists of consecutive vertices, therefore by proposition 3.4 the diameter of Z_{k-a-1} did not increase.

By our choice of U, Z_{k-a} is maximal.

If $diam(Z_{k-a} \bigcup U) < B - 1$ then, since the graph is connected, there is a vertex $v^* < s_{k-a} - 1$, in $Z_{k-a-1} - U$, that is adjacent to s_{k-a}, such that $diam(Z_{k-a} \bigcup \{v : v^* \leq v < s_{k-a}\}) \leq B$ and $diam(Z_{k-a-1} - U - \{v : v^* \leq v < s_{k-a}\}) \leq B$, but this contradicts our choice of x^*. Therefore $diam(Z_{k-a}) \geq B - 1$.

Therefore there is an optimal partition that has a larger number of consecutive sets that satisfy the theorem, which contradicts our choice of $\mathcal{Z}$. ■

We now present a sequential algorithm that uses the result of the above theorem and DFS to find a non-overlapping partition. First define the following.

> **Definition 3.3:** $prev(u) = \min_{H(v)} v$ *is adjacent to* u. In the case of a tie we choose $prev(u)$ to be the vertex with the smallest index among those satisfying the above statement.
>
> Let $prev^0(u) = u$, and $prev^l(u) = prev(prev^{l-1}(u))$ for $l > 0$.

Note the analogy between the definition of $prev(v)$ and that of $next(v)$ (definition 3.1); $prev(v)$ is the neighbor of v that extends the most to the left, and $next(v)$ is that neighbor that extends the most to the right.

We now prove the following propositions pertaining to our algorithm.

Proposition 3.6 *Given vertices u and w, $w > u$, $(u, w) \notin E$, there is a path $P(u, w) = \{u = v_{i_0}, v_{i_1}, v_{i_2}, ...v_{i_p}, w = v_{i_{p+1}}\}$ in $\mathcal{P}(u, w)$, such that $v_{i_{j-1}} = prev(v_{i_j})$, for all $j > 1$.*

Proof : Reverse the order of the vertices of $G = (V, E)$, i.e. let $g(v_{n-i}) = v_{i+1}$, for all i, $i = n-1, \ldots, 0$, and let $V' = \{g(v_{n-i}), i = n-1, \ldots, 0\}$ and $E' = \{(g(v_j), g(v_i)) : (v_i, v_j) \in E\}$, then $G' = (V', E')$ is isomorphic to G. And, for all v in V, $prev(v)$ in V and $next(g(v))$ in V' correspond to the same interval. Consider $P(g(u), g(w))$, the result now follows by proposition 3.1. ■

Now the next proposition is easy to see.

Proposition 3.7 *For all u and v, $u > v$, $dist(u, v) \leq B$ if and only if v is adjacent to $prev^l(u)$ for some $l \leq B - 1$.*

Proof: Consider a path between u and v that satisfies proposition 3.6. ■

The next proposition is a direct consequence of the above, and shows how far we should go to build a maximal set. Let Z_i be a set in an optimal partition $\mathcal{Z}$.

Proposition 3.8 *Given $t(Z_i)$, Z_i is maximal if and only if $s(Z_i)$ is the smallest index vertex smaller than or equal to $prev^{B-1}(t(Z_i))$ such that all vertices v, $s(Z_i) \leq v \leq prev^{B-1}(t(Z_i))$ are adjacent to $prev^{B-1}(t(Z_i))$.*

We now give an overview of the algorithm.

The algorithm we present greedily finds sets of nearly equal diameter as in theorem 3.4.

Let G_1 be the subgraph of G not partitioned so far.

The algorithm starts at $u_0 = t(G_1)$ and greedily finds a set that extends as far as possible from u_0 (a maximal set), following the path u_0, $prev(u_0)$, $prev(prev(u_0)), \ldots$ (see proposition 3.6).

To find a set that extends the farthest, DFS is used, where vertices of the DFS path are u_0, $prev(u_0)$, $prev(prev(u_0)), \ldots$ (see proposition 3.6), up to $prev^{B-1}(u_0)$, then we add all vertices v such that $v \leq v^*$ where v^* satisfies the conditions in proposition 3.8, i.e. v^* becomes the first vertex of the current set.

On returning from the DFS subroutine v^* is stored in array T as the vertex that marks the beginning of the set (in the algorithm the vertex $u_0 + 1$ corresponds to v^*). Therefore at the end we have the partition: $\mathcal{Z} = \{Z_1, Z_2, \ldots, Z_k\}$, where $Z_i = \{T(i), T(i)+1, \ldots, T(i+1)-1\}$.

We assume that Λ is the *null* vertex.

Algorithm 3.2 Algorithm PVD-INTERVAL(G, B, k, V)

Input: *Integer B and $G = (V, E)$ where G is an interval graph , $V = \{v_1, \ldots, v_n\}$.*
Output: *The cardinality k of the optimal partition, $k = |\mathcal{Z}|$.*

begin

1. **for all** $v \in V$ **do**
   unmark v
   endfor
2. $T(index) = \Lambda$, $1 \leq index \leq n$
3. $u_0 = t(G)$
4. $index = 1$
5. **while** $u_0 \neq \Lambda$ **do**
   Call DFS(G, B, u_0)
   $T(index) = u_0 + 1$
   $index = index + 1$
   enddo
6. $k = index - 1$
   end

Procedure DFS(G, B, u_0)
begin

1. $v = u_0$
2. $it = 0$
3. **while** $v \neq \Lambda$ *and* $it \leq B - 1$ **do**
   mark v
   $curr = v$
   $v = prev(v)$
   $it = it + 1$
   enddo
   /Note that $curr = prev^{B-1}(v)$/
4. $u_0 = \max_i v_i < curr, v_i$ *is not adjacent to* $curr$
   end

Theorem 3.5 *Algorithm 3.2 computes the cardinality, k, of the optimal partition that satisfies theorem 3.4 in $\mathcal{O}(n)$ time.*

Proof: To prove the correctness of the algorithm it suffices to prove that the non-overlapping partition $\mathcal{Z} = \{\{T(1), T(1)+1, \ldots\}, \{T(2), \ldots\}, \ldots\}$ is the optimal partition that satisfies theorem 3.4.

We prove the above by induction on the cardinality of $\mathcal{Z}$. If $|\mathcal{Z}| = 1$ then it is clear that the algorithm performs correctly (see proposition 3.8).

Assume the hypothesis is true for all interval graphs with optimal partitions of cardinality smaller than k. Let G be an interval graph that has an optimal partition of cardinality k. Let $\mathcal{Z}$ be the partition for G produced by the algorithm. Let $\mathcal{Z}^*$ be the partition that satisfies theorem 3.4.

Let $\mathcal{Z} = \{Z_1, Z_2, \ldots, Z_{kk}\}$, and $\mathcal{Z}^* = \{Z_1^*, \ldots, Z_k^*\}$, where $kk > k$.

Suppose $Z_{kk-i} = Z_{k-i}^*$, for all i less than a largest integer a. Let G' be the subgraph induced by the sets $Z_{k-j}, j \geq a$. G' is also the subgraph induced by all $Z_{k-j}^*, j \geq a$. Therefore G' has an optimal partition of cardinality $k - a$ that satisfies the theorem. If $a > 0$, then by the induction hypothesis it must be true that $\mathcal{Z}$ has cardinality k.

Therefore suppose that $a = 0$. But since any maximal set that starts at $t(G)$ is unique (see proposition 3.5) and the algorithm outputs a maximal set Z_k that satisfies proposition 3.8, then it must be true that $Z_k = Z_k^*$.

This completes the correctness proof. We now prove the complexity of the algorithm.

First recall that to compute $prev(v)$ for a given v, at most $\mathcal{O}(n)$ time is needed. Moreover, since the intervals are already sorted then the same amount of time is needed to compute $t(G)$.

Each step in the main procedure prior to the loop (step 4) requires linear time.

All steps in the loop, excluding for now the call for DFS, require constant time.

We now show that each vertex is scanned at most twice.

Let u_0^i be the value of u_0^i upon entry to the DFS subroutine for the ith time. Each vertex v, $u_0^{i+1} < v \leq u_0^i$ is scanned once in the DFS subroutine, either in the main loop or in the last statement (to find u_0^{i+1}). These vertices will not be scanned in subsequent executions of DFS. Therefore the only vertices scanned more than once are u_0^i, $i \geq 2$. Since $u_0^{i+1} < u_0^i$, therefore each u_0^i is scanned at most twice, once in the last statement during the $i-1$st execution of DFS and the second time in the loop of DFS during the ith execution of DFS.

Therefore every vertex in G is scanned at most twice by DFS, and the total time of the algorithm is $\mathcal{O}(n)$. ■

Chapter 4

Partitioning Biconvex Graphs: Vertex Ordering

4.1 Overview

In the next three chapters we study the problem $PVMF$ (partitioning vertices to minimize monotone function) on biconvex graphs. We already proved that the special case PVD (partitioning vertices to minimize maximum diameter) is hard on bipartite graphs in general, for all fixed values of the maximum diameter greater than one (see chapter 2 of this thesis).

Our objective now is to devise an NC parallel algorithm for the problem $PVMF$ on biconvex graphs. In this chapter we prove some useful properties of biconvex graphs that will enable us to devise such an algorithm. The algorithm is presented in chapter 6 along with a sequential algorithm for PVD on bipartite permutation graphs. The most useful property we prove is one that deals with the ordering of vertices. We prove that we can reorder the vertices of a biconvex graph so that they have, what we call, a *biconvex straight ordering.* This ordering is a generalization of the strong ordering achievable for bipartite permutation graphs (see Appendix B). We also prove some lemmas pertaining to the diameter and distances in biconvex graphs. In the last section of this chapter we devise some useful parallel algorithms that will be used by the algorithm for $PVMF$.

Recall problem $PVMF$.

> **Problem $PVMF$: Partitioning Vertices to Minimize Monotone Function:** Given a graph $G = (V, E)$, a monotone diameter function f, and non negative integers k and B, is there a partition $\mathcal{Z}$ of V into k or fewer sets, such that $f(\mathcal{Z}) \leq B$?

The definition of monotone diameter functions is given in the introductory chapter (definition 1.1).

Problem PVD (Partitioning Vertices to Minimize Maximum Diameter) is a special case of this problem obtained by setting $\bullet$ to max in the definition of $f(\mathcal{Z})$.

As discussed in chapter 2, maximum matching on bipartite graphs may be seen as a special case of $PVMF$ ($\bullet$ = max, and $B = 1$). Parallel matching on general graphs is in *Random NC (RNC)* [52]. When the input graph is convex the problem falls in the class NC (see [23] and [24] for parallel algorithms and [38] for a sequential algorithm). We are unaware of any parallel algorithms for other special cases of $PVMF$ on convex graphs.

Biconvex graphs arise in some industrial applications, see for example [38]. Convex graphs, which include biconvex graphs, arise in scheduling problems [23]. These graphs possess some good algorithmic properties with regards to vertex ordering.

We first review definitions of the graph families studied in this and the next two chapters.

A bipartite graph $G = (X, Y, E)$ is *biconvex* if X and Y can be ordered so that for every vertex v in $X \bigcup Y$ neighbors of v occur consecutively in the ordering [16].

Convex graphs are a superset of biconvex graphs; in these graphs convexity is required for the neighbors of either the vertices of X or those of Y (see [16] and [15]).

Convex graphs are a proper subset of *chordal bipartite* graphs; these are bipartite graphs with no induced chordless cycle of length greater than four [40].

Bipartite Permutation Graphs, or *BPGs*, are a subset of biconvex graphs. In addition to the biconvexity condition, there is an ordering of the vertices of these graphs that has the *enclosure property*. The enclosure property is described as follows [71], for all u, v in $X \bigcup Y$, if $N(u) \subset N(v)$ then vertices in $N(v) - N(u)$ occur consecutively. Alternatively, *BPGs* are characterized as those bipartite graphs whose vertices have a *strong ordering*. A strong ordering is described as follows [71], for all x_{i1}, x_{j1} in X and y_{i2}, y_{j2} in Y, if (x_{i1}, y_{i2}) is in E, (x_{j1}, y_{j2}) is in E, $x_{i1} < x_{j1}$, and $y_{j2} < y_{i2}$ then (x_{i1}, y_{j2}) is in E and (x_{j1}, y_{i2}) is in E.

In summary the aforementioned graph families bear the relationship: $BPG \subset Biconvex \subset Convex \subset Chordal\ Bipartite \subset Bipartite$. Figure 4.1 shows examples of these graph families, where each example graph is not contained in the lower family.

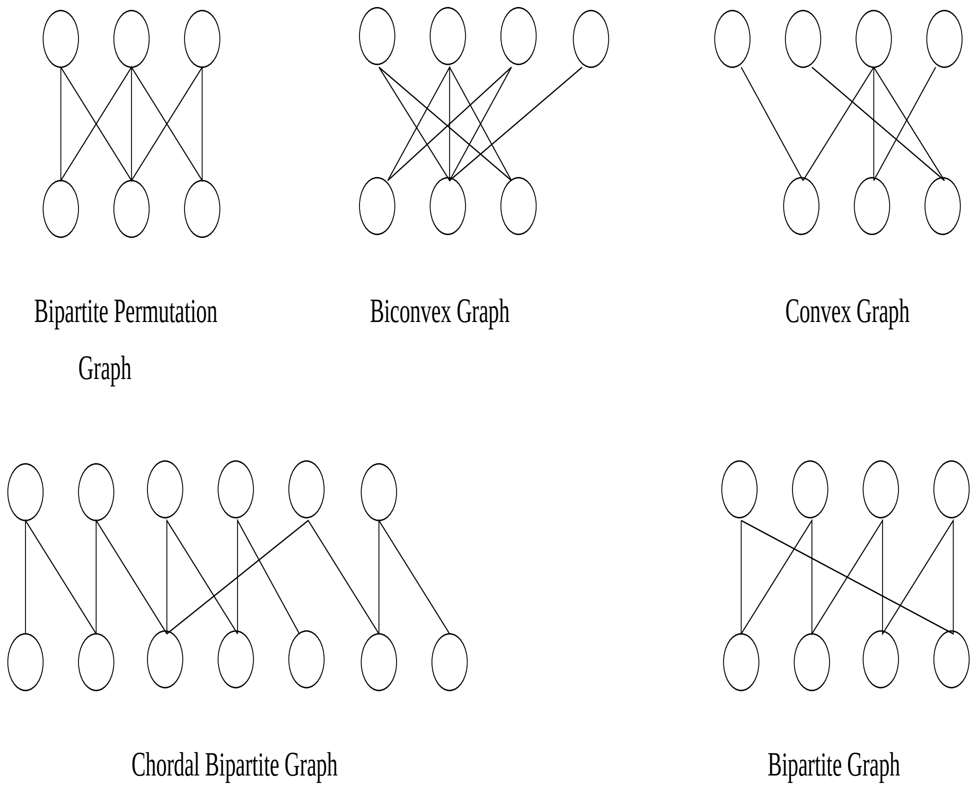

Figure 4.1: Various special bipartite graphs

In the next section we present necessary definitions and notation. Section 4.3 contains the main theorem that deals with the new suggested vertex order, and various other proofs about distances and diameters that we will mainly use in the next two chapters. In the last section we present an efficient parallel algorithm to obtain the new ordering, and show how to compute distances and diameters efficiently in parallel.

4.2 Definitions and Notation

Let $G = (X, Y, E)$ be a biconvex graph. Let $n = |X \bigcup Y|$ and $m = |E|$. Pictorially, vertices of X will be at the bottom and the smallest index vertex of either X or Y will be at the leftmost end of the graph. Define the following.

For any set $V \subseteq X \bigcup Y$, let $V^X = V \bigcap X$ and $V^Y = V \bigcap Y$.

For any vertex v in $X \bigcup Y$, let $N(v) = \{u : u \ is \ adjacent \ to \ \ v\}$.

Let $\mathcal{P}(u, v/W)$ be the set of all shortest paths between vertices u and v in the subgraph, $< W >$, of G induced by $W \subseteq X \bigcup Y$. Let $\mathcal{P}(u, v) = \mathcal{P}(u, v/X \bigcup Y)$.

Let $P(u, v/W)$ be the set of vertices in an arbitrary element of $\mathcal{P}(u, v/W)$. Let $P(u, v) = P(u, v/X \bigcup Y)$.

Let the distance between vertices u and v in the subgraph, $< W >$, induced by vertices in $W \subseteq (X \bigcup Y)$, be $dist(u, v/W) = |P(u, v/W)| - 1$, for any $P(u, v/W)$ in $\mathcal{P}(u, v/W)$.

Let the diameter of the set of vertices W, or of the subgraph, $< W >$, of G induced by W, be $diam(W) = \max_{u,v \in W} dist(u, v/W)$.

$\mathcal{Z}$ is an *optimal partition* for G if $f(\mathcal{Z}) \leq B$, and there is no other such partition for G of cardinality smaller than $k = |\mathcal{Z}|$.

For all $Z_i \in \mathcal{Z}$, let s_i^X and s_i^Y be the first vertices in Z_i^X and Z_i^Y, respectively. Let t_i^X and t_i^Y be the last vertices.

For all $Z_i \in \mathcal{Z}$, let the *range* of Z_i be $R(Z_i) = R_i = R_i^X \bigcup R_i^Y$, where $R(Z_i^X) = R_i^X = \{x : s_i^X \leq x \leq t_i^X\}$ and $R(Z_i^Y) = R_i^Y = \{y : s_i^Y \leq y \leq t_i^Y\}$. Figure 4.2 shows a set Z_i, its ranges and its various special vertices.

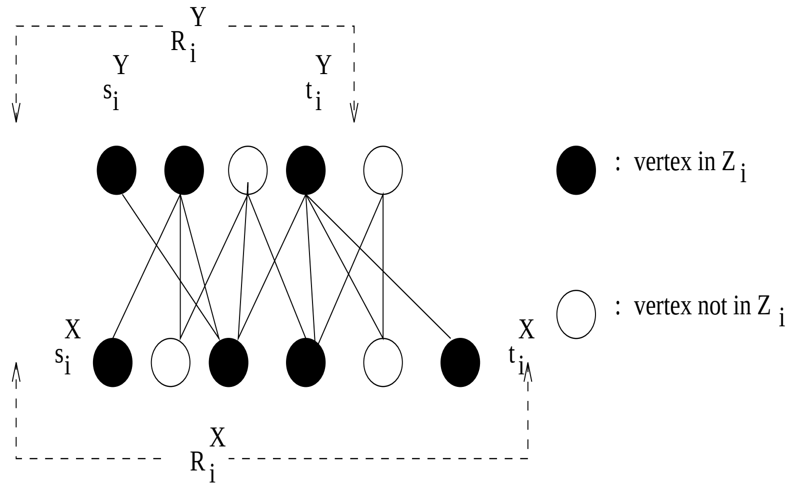

Figure 4.2: A set Z_i, its ranges, and various special vertices

Define the relation $<_S$ on members of $\mathcal{Z}$, so that $Z_i <_S Z_j$ if $s_i^X < s_j^X$. If Z_i is a cardinality one set, say $\{s_i^Y\}$, then $Z_i <_S Z_j$ if $Z_j^Y \neq \phi$ and $s_i^Y < s_j^Y$. If $Z_i^Y = \phi$ and $Z_j^X = \phi$ then Z_i and Z_j are unrelated.

Define the relation $<$ on subsets of $X \bigcup Y$, $V_1 < V_2$ (or $V_2 > V_1$), if for every vertex v_1 in V_1^X (or V_1^Y) and every vertex v_2 in V_2^X (resp. V_2^Y), $v_1 < v_2$. $v < V_1$ will be used instead of $\{v\} < V_1$. We will use $v_1 < V_2$ instead of $\{v_1\} < V_2$. If V_1 and V_2 are unrelated we will use the notation $V_1 <> V_2$.

For all $x \in X$ (or $y \in Y$), let $prev(x/W) = \min_j y_j$ *is adjacent to* x *in* $< W >$, and $next(x/W) = \max_j y_j$ *is adjacent to* x *in* $< W >$. We will use the notation $prev(x)$ and $next(x)$ when $W = X \bigcup Y$ or when no confusion might arise.

4.3 Some Properties of Biconvex Graphs

A biconvex graph is one for which there exists a biconvex ordering. Let $G = (X, Y, E)$ be a biconvex graph where $X = \{x_1, \ldots, x_p\}$, $Y = \{y_1, \ldots, y_q\}$, and $X \bigcup Y$ define a biconvex ordering. We now prove some properties of biconvex graphs that will be useful to design an algorithm for $PVMF$. We will sometimes omit results implied by the symmetry in biconvex graphs. Biconvex graphs are symmetric in the following two ways. If $G = (X, Y, E)$ is a biconvex ordering of a biconvex graph, then the following are also biconvex orderings for G: $G = (Y, X, E)$ (by horizontal symmetry), and $G = (X', Y, E)$ where $X' = \{x_p, \ldots, x_1\}$ (by vertical symmetry).

The first lemma in this section shows that for every path $P(u, v)$ in $\mathcal{P}(u, v)$, vertices of $P^X(u, v)$ and vertices of $P^Y(u, v)$ are linearly ordered.

Lemma 4.1 (Order of Vertices in Paths) *Consider a shortest path, $P(u, v)$, between any two vertices u, v in a biconvex graph G. Let vertices of $P^X(u, v)$ and vertices of $P^Y(u, v)$ be ordered as in $P(u, v)$, $P^X(u, v) = \{x_1, x_2, \ldots, x_l\}$ and $P^Y(u, v) = \{y_1, y_2, \ldots, y_o\}$. Then $x_j < x_{j+1}$ for all $x_j \in P^X(u, v)$ or $x_j > x_{j+1}$ for all $x_j \in P^X(u, v)$, and $y_j < y_{j+1}$ for all $y_j \in P^Y(u, v)$ or $y_j > y_{j+1}$ for all $y_j \in P^Y(u, v)$.*

Proof : Assume not. Let $x_a > x_{a+1}$ and $x_j < x_{j+1}$ for all $j < a$. Then, by convexity, $x_{a-1} < x_{a+1}$ implies that x_{a+1} is adjacent to the common neighbor of x_{a-1} and x_a in $P(u, v)$ and x_a is not needed, otherwise $x_{a-1} > x_{a+1}$ implies that x_{a-1} is adjacent to the common neighbor of x_{a+1} and x_a in $P(u, v)$ and again x_a is not needed. Both cases contradict the fact that $P(u, v)$ is a shortest path between u and v. Similarly we can show that vertices of $P^Y(u, v)$ have to be linearly ordered. ∎

Definition 4.1: $P(u,v)$ is called an *S-path* (straight path) if the order of the vertices of $P^X(u,v)$ is the same as that of the vertices of $P^Y(u,v)$ (see figure 4.3 a). Otherwise it is called a *C-path* (crooked path) (see figure 4.3 b).

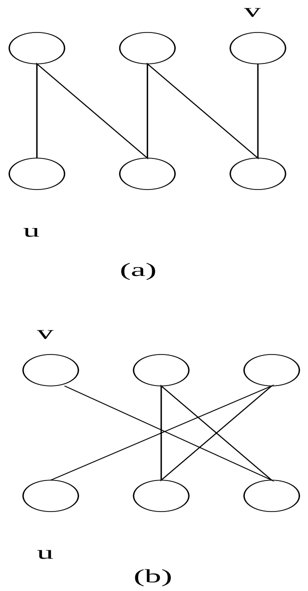

Figure 4.3: **(a) An S-path between u and v**
(b) A C-path between u and v

The next definition is necessary for the proof of the next lemma, which in turn is crucial for the proof of the main theorem of this chapter.

Definition 4.2: The edges $e_i = (x_{i1}, y_{i2})$ $e_j = (x_{j1}, y_{j2})$ are said to *cross* if $x_{i1} < x_{j1}$ and $y_{i2} > y_{j2}$.

Lemma 4.2 (Crossing Edges) *If (x_i, y_j) is an edge in E and there are edges (x_a, y_d) and (x_b, y_c) that cross such that $x_i \leq x_a < x_b$ and $y_j \leq y_c < y_d$ (or $x_a < x_b \leq x_i$ and $y_c < y_d \leq y_j$) then (x_a, y_c) (respectively (x_b, y_d)) is an edge in E.*

Proof : Assume there are vertices $x_i, x_a, x_b, y_j, y_c,$ and y_d such that $x_i \leq x_a < x_b$, $y_j \leq y_c < y_d$, $(x_a, y_d), (x_b, y_c)$, and (x_i, y_j) are edges in E but (x_a, y_c) is not in E.

If $x_i = x_a$ or $y_j = y_c$ then by convexity x_a is adjacent to y_c.

Therefore suppose that $x_i < x_a$ and $y_j < y_c$. If y_c has a neighbor smaller than x_a then y_c is adjacent to x_a (by convexity). As well if y_j has a neighbor larger than x_a then x_a is adjacent to both y_j and y_c (by biconvexity). Therefore all neighbors of y_j are smaller than x_a and all neighbors of y_c are larger than x_a. Therefore the path between y_j and y_c is an S-path. But now there is a vertex y on that path, $y < y_c$, that has a neighbor smaller than x_a and a neighbor greater than x_a, and thus y is adjacent to x_a and x_a is adjacent to y_c (by biconvexity). ∎

Corollary 4.1 *If (x_1, y_1) and (x_p, y_q) are edges in E then G is a strongly ordered bipartite permutation graph.*

We next define an ordering of the vertices that generalizes the strong ordering for bipartite permutation graphs. Then we prove that biconvex graphs have such an ordering.

> **Definition 4.3:** A *straight ordering* (*S-ordering* for short) of the vertices of a bipartite graph $G = (X, Y, E)$ is an ordering that satisfies the following. If $e_i = (x_{i1}, y_{i2})$ and $e_j = (x_{j1}, y_{j2})$ are edges in E that cross then at least one of (x_{i1}, y_{j2}) and (x_{j1}, y_{i2}) is in E.

> **Definition 4.4:** A *biconvex S-ordering* of the vertices of a bipartite graph $G = (X, Y, E)$ is an S-ordering that preserves biconvexity.

In the next theorem we prove that there is a biconvex S-ordering of the vertices of G. We reorder subsets of the vertices of G at most three times. The first reordering is on vertices of Y. This operation, called **ReverseY**, merely reverses the order of the vertices of Y. If necessary we then reorder a subset of X and/or a subset of Y, we call these operations **ExchangeX** and **ExchangeY**, respectively. The proof is constructive and will later be used as a basis for a parallel algorithm that produces a biconvex S-ordering for G.

Theorem 4.1 (New Biconvex Ordering) *Let $G = (X, Y, E)$ be a biconvex graph, G has a biconvex S-ordering.*

Proof : Let X and Y have a biconvex ordering. Suppose the claim is not true, then there are vertices x_a, x_b, y_c, y_d such that $x_a < x_b$, $y_c < y_d$, and (x_a, y_d) crosses (x_b, y_c) but neither (x_a, y_c) nor (x_b, y_d) is in E. Let $e = (x_i, y_j)$ be some other edge in G. If $x_i \le x_a$ and $y_j \le y_c$ (or $x_i \ge x_b$ and $y_j \ge y_d$) then by lemma 4.2 and convexity x_a is adjacent to y_c (respectively x_b is adjacent to y_d). Thus, no edge has both endpoints smaller than or equal to x_a or y_c (or greater than or equal to x_b or y_d). Therefore any edge in G crosses one or both of (x_a, y_d) and (x_b, y_c). We proceed successively to the end of the proof.

(a) We first prove that there are vertices x_a', x_b', y_c', y_d' such that (x_a', y_d') and (x_b', y_c') are in E, (x_a', y_c') and (x_b', y_d') are not in E, $x_a' < x_b'$ and $y_c' < y_d'$, and all edges of the graph have at least one endpoint in the set $\{x, y : x_a' \le x \le x_b', y_c' \le y \le y_d'\}$.

If all edges in G have an endpoint in the set $\{x, y : x_a \le x \le x_b, y_c \le y \le y_d\}$, then $x_a' = x_a, x_b' = x_b, y_c' = y_c$, and $y_d' = y_d$.

Else let (x_a', y_d') and (x_b', y_c') be edges that have no endpoint in the set $\{x, y : x_a < x < x_b, y_c < y < y_d\}$, such that $x_1 \le x_a' \le x_a$, x_a' has the smallest index among all such vertices, and $x_b \le x_b' \le x_p$, x_b' has the largest index among all such vertices.

This edge pair satisfies the requirements (see figure 4.4 for an illustration).

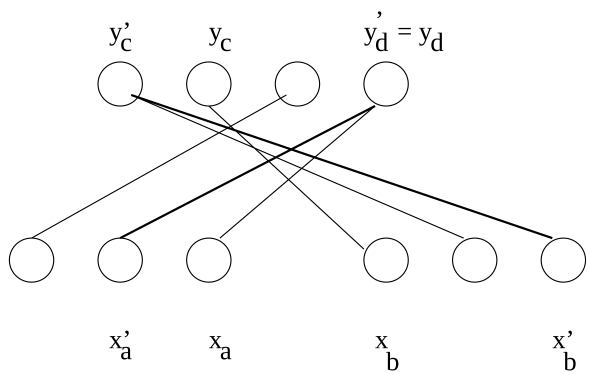

Figure 4.4: Every edge in G has some endpoint in the set $\{x, y : x_a^{'} \le x \le x_b^{'}, y_c^{'} \le y \le y_d^{'}\}$

If $x_a^{'}$ has a neighbor smaller than $y_c^{'}$ it will be adjacent to $y_c^{'}$ and consequently, by biconvexity, x_a and y_c will be adjacent. Similarly $y_c^{'}$ has no neighbor smaller than or equal to $x_a^{'}$. Therefore the paths $P(x_a^{'}, y_c^{'})$ and $P(x_b^{'}, y_d^{'})$ are C-paths.

(b) We now perform the first reordering of the vertices.

ReverseY: Reverse the order of vertices of Y.

Since we still have a biconvex ordering, then by corollary 4.1 the subgraph G_1, induced by the set $\{x, y : x_a^{'} \le x \le x_b^{'}, y_d^{'} \le y \le y_c^{'}\}$ is now a strongly ordered bipartite permutation graph, and edges with both endpoints in G_1 satisfy the claim of the theorem. Every vertex not in G_1 is adjacent to some vertex in it and has no neighbors outside of it (by our choice of $\{x_a^{'}, x_b^{'}, y_c^{'}, y_d^{'}\}$). The path $P(x_a^{'}, y_c^{'})$ is now an S-path.

Assume some vertices of $P(x_a^{'}, y_c^{'})$ are not in G_1. Let S be a set that contains such vertices. Since $P(x_a^{'}, y_c^{'})$ is an S-path, therefore $S^Y < G_1$ or S^Y is empty, and $S^X > G_1$ or S^X is empty . Suppose S^Y is not empty. Let vertices in S^Y be $y_1^{'} < y_2^{'} < \cdots y_i^{'}$ but $y_i^{'} < y_d^{'}$ and $y_{i+1}^{'} > y_d$ implies that $y_d^{'}$ may replace all vertices $y_1^{'}, \ldots, y_i^{'}$ in $P(x_a^{'}, y_c^{'})$. A similar argument applies if $S^X \ne \phi$. Therefore there is a path between $x_a^{'}$ and $y_c^{'}$ that is contained in G_1 and G_1 is connected (since every vertex in G_1 is adjacent to some vertex in that path by biconvexity).

(c) Assume now that there are still edges (x_{i1}, y_{i2}) and (x_{j1}, y_{j2}) that cross with (x_{i1}, y_{j2}) and (x_{j1}, y_{i2}) not in E; at least one endpoint of each edge has to be in G_1.

If $x_{i1} < x_a^{'}$, $x_{j1} > x_b^{'}$, and $y_d^{'} < y_{j2} < y_{j1} < y_c^{'}$, then we prove that by rearranging a subset of the vertices of G we get the desired result. First observe the following.

(i) For all u, v such that $v < u \le x_{i1}$ or $x_{j1} \le u < v$, $N(v) \subseteq N(u)$ (since every neighbor, w, of v is in G_1 and has a neighbor in G_1, therefore by convexity w is also adjacent to u). Therefore there is a largest index vertex $x_{a1} \ge x_{i1}$ and a smallest index vertex $x_{b1} \le x_{j1}$ such that there are edges (x_{a1}, y_{d1}) and (x_{b1}, y_{c1}) that cross and do not satisfy the claim (see figure 4.5), $N(x) \subseteq N(x_{a1}) \forall x \le x_{a1}$, and $N(x) \subseteq N(x_{b1}) \forall x \ge x_{b1}$.

Let $X_L = \{x : x_1 \le x \le x_{a1}\}$, $X_R = \{x : x_{b1} \le x \le x_p\}$ (see figure 4.5). All neighbors of vertices in these sets are in G_1 and it must be true that $prev(x_{b1}) < prev(x_{a1})$ and $next(x_{b1}) < next(x_{a1})$ (otherwise one of the edges (x_{a1}, y_{c1}) and (x_{b1}, y_{d1}) would be present contradicting our choice of (x_{a1}, y_{d1}) and (x_{b1}, y_{c1})).

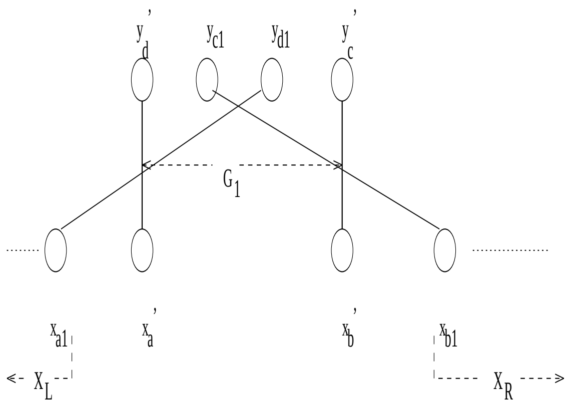

Figure 4.5: Edges (x_{a1}, y_{d1}) and (x_{b1}, y_{c1}) do not satisfy the claim

Next observe the following.

(ii) For all $y_i, y_j \in Y$ such that y_i is adjacent to x_{a1} but not to x_{b1} and y_j is adjacent to x_{b1} but not to x_{a1}, it is true that $y_i > y_j$. Since both y_i and y_j have some neighbor in G_1 therefore y_i is adjacent to x'_a and y_j is adjacent to x'_b. Thus by convexity y_i and y_j are adjacent to all vertices in G_1^X. Let x be such that $x_{a1} < x < x'_a$, by convexity (x, y_i) is in E, and by our choice of x_{a1} (x, y_j) is in E. Therefore all vertices $y \in N(x_{a1}) \bigcup N(x_{b1})$ are adjacent to all vertices in $X - X_L - X_R$.

(d) We now perform the second reordering of the vertices.

ExchangeX: We rearrange vertices of $X_L \bigcup X_R$ as follows. For every vertex x_{a1-i} we place x_{a1-i} in the $(b1+i)$th place, and place x_{b1+i} in the $(a1-i)$th place (as though vertices of X_R and X_L are rotated around G_1). Therefore neighborhood sets now satisfy $prev(x_{a1}) < prev(x_{b1})$ and $next(x_{a1}) < next(x_{b1})$. We note that the ordering is still biconvex since for all values of i, all vertices y in $N(x_{a1-i})$ are adjacent to all vertices in $(X - X_L - X_R) \bigcup \{x : x_{a1-i} \le x \le x_{a1}\}$, and all vertices y in $N(x_{b1+i})$ are adjacent to all vertices in $(X - X_L - X_R) \bigcup \{x : x_{b1} \le x \le x_{b1+i}\}$ (see observations (i) and (ii) above).

(e) Now $prev(x_{b1}) > prev(x_{a1})$, all neighbors of vertices larger than x_{b1} are in $N(x_{b1})$, and all neighbors of vertices smaller than x_{a1} are in $N(x_{a1})$. Thus we eliminated all edge pairs e_i, e_j, such that $e_i = (x_{i1}, y_{i2})$, $e_j = (x_{j1}, y_{j2})$, $x_{i1} < x'_a$, $x_{j1} > x'_b$, $y'_c < y_{j2} < y_{i2} < y'_d$, and both (x_{i1}, y_{j2}) and (x_{j1}, y_{i2}) are not in E. We also note that we have not introduced any edge pairs e_i, e_j, such that $e_i = (x_{i1}, y_{i2})$, $e_j = (x_{j1}, y_{j2})$, $y_{i2} < y'_c$, $y_{j2} > y'_d$, $x'_a < x_{j1} < x_{i1} < x'_b$, and both (x_{i1}, y_{j2}) and (x_{j1}, y_{i2}) are not in E. However such a pair might already exist. In this case we define sets Y_R and Y_L, similar to sets X_R and X_L defined above, and rearrange their vertices analogously, we call that operation: **ExchangeY**.

(f) If there are still edges (x_{a1}, y_{d1}) and (x_{b1}, y_{c1}) that cross with (x_{a1}, y_{c1}) not in E and (x_{b1}, y_{d1}) not in E, then at least two of $\{x_{a1}, y_{d1}, x_{b1}, y_{c1}\}$ are in G_1 and at least one of $\{x_{a1}, y_{d1}, x_{b1}, y_{c1}\}$ is not in G_1. Thus we have one of the situations in figure 4.6 (ignoring symmetric situations). In any of these situations the edge $(x_b^{'}, y_c^{'})$ has both endpoints larger than or equal to y_{d1} or x_{b1}. Therefore by lemma 4.2 the edge (y_{d1}, x_{b1}) exists. ■

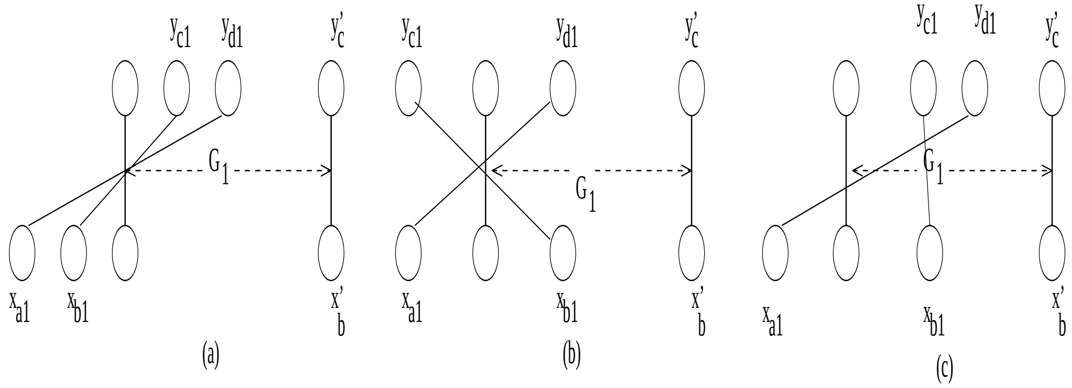

Figure 4.6: Three cases of crossing edges

In what follows we assume that G has a biconvex S-ordering.

Corollary 4.2 *All shortest C-paths in G have length less than or equal to three. Furthermore the paths $P(x_1, x_p)$, $P(y_1, y_q)$, $P(x_1, y_q)$, and $P(y_1, x_p)$ are all S-paths, and each of $dist(x_1, y_1)$ and $dist(y_1, y_q)$ is either one or three.*

Proof : Let $P(u, v)$ be a C-path of length greater than three, $P^X(u, v) = \{x_1^{'}, x_2^{'}, \ldots, x_l^{'}\}$ and $P^Y(u, v) = \{y_1^{'}, y_2^{'}, \ldots, y_o^{'}\}$, where $x_1^{'} < x_2^{'} < \cdots$ and $y_1^{'} > y_2^{'} > \cdots$. Then the edges $(x_1^{'}, y_1^{'})$ and $(x_l^{'}, y_o^{'})$ cross. Therefore either $x_1^{'}$ and $y_o^{'}$ are adjacent or $y_1^{'}$ and $x_l^{'}$ are adjacent and in both cases $dist(u, v) \leq 3$. Since x_1, y_q are the end vertices of the graph then $P(x_1, y_q)$ (as well as $P(y_1, x_p)$) is an S-path. ■

Corollary 4.3 *If $G = (X, Y, E)$ is a strongly ordered bipartite permutation graph then all shortest paths in G are S-paths.*

We now study the structure of special biconvex graphs of bounded diameter.

Lemma 4.3 (Structure of Diameter Three Graphs) *Let $diam(G) = 3$ then there is an edge (x^*, y^*) in E such that x^* is adjacent to all vertices of Y and y^* is adjacent to all vertices of X.*

Proof : Since $diam(G) = 3$ then $dist(x_1, x_p) = dist(y_1, y_q) = 2$, there is a vertex y^* adjacent to x_1 and x_p, and there is a vertex x^* adjacent to y_1 and y_q. Therefore, by biconvexity, y^* is adjacent to all x in X and x^* is adjacent to all y in Y. ■

We now define some special biconvex graphs of bounded diameter that have special structure.

Definition 4.5: Special Biconvex Graphs. Let $G = (X, Y, E)$ be a biconvex graph such that $X = \{x_1, \ldots, x_p\}$ and $Y = \{y_1, \ldots, y_q\}$. We define the following graphs.

Star : $G = (X, Y, E)$ is a star if it has diameter less than or equal to two and $|X| = 1$ (*X-star*) or $|Y| = 1$ (*Y-star*).

Estar : $G = (X, Y, E)$ is an Estar (edge star) if it has diameter three. If x_p and y_q are adjacent to all vertices in G then G is an *R-Estar* (right Estar), if x_1 and y_1 are adjacent to all vertices in G then G is an *L-Estar* (left Estar).

The next definitions show the special structure of the above graphs. See figure 4.7 for examples of these graphs.

Definition 4.6: G is an X-star (or a Y-star), then x_1 (respectively y_1) is called the *center of the star*. Note that each star has a unique center, unless it has cardinality two.

Definition 4.7: Let G be an Estar, let (x^*, y^*) be an edge whose endpoints are adjacent to all other vertices in G, then (x^*, y^*) is called a *center of the Estar*.

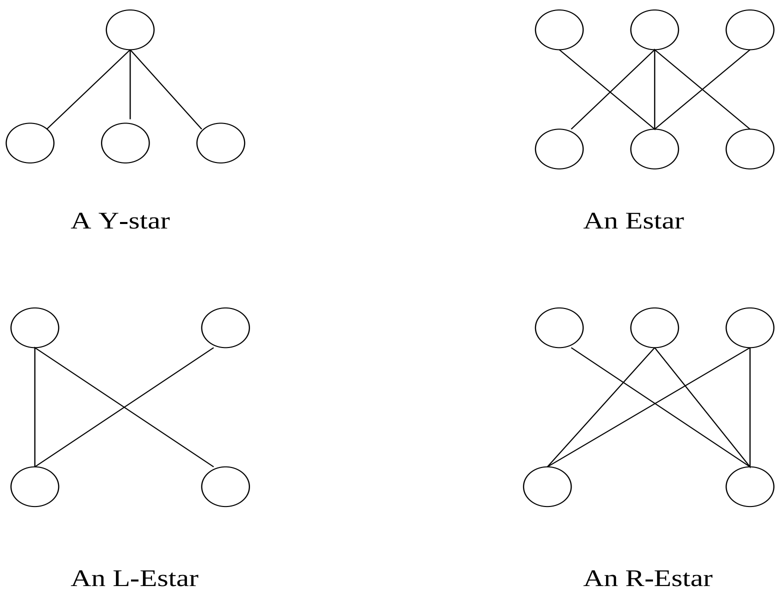

Figure 4.7: Some special biconvex graphs

Let $\mathcal{Z} = \{Z_1, \ldots Z_i, \ldots Z_k\}$ be an optimal partition for the biconvexly S-ordered graph $G = (X, Y, E)$. Let Z_i be any set in $\mathcal{Z}$. The next few lemmas are essential for the next two chapters. Expectedly enough they deal with diameters of general and special sets. In the next lemma we show how the start and end vertices of a subgraph determine the diameter of the subgraph. Subsequent lemmas depend on that result. In lemma 4.6 we prove that if removing one vertex from a set increases its diameter then adding neighbors of that vertex to that set will not increase the diameter. In lemma 4.5 we show how an R-Estar or an L-Estar may be extended to a general Estar. In the last lemma (lemma 4.7) of this section we show how under certain conditions we can extract a subset of any set that has a diameter not larger than the diameter of the original set.

Lemma 4.4 (End Vertices and Diameter) *For every* u *and* v *in* R_i, $dist(u,v) \le diam(\{s_i^X, s_i^Y, t_i^X, t_i^Y\}/$ *where* $diam(U/W) = \max_{u,v\in U} dist(u, v/W)$ *for* $U \subseteq W \subseteq X \bigcup Y$.

Proof : Let u and v be two nonadjacent vertices in G. Let $u \in R_i^X$ and $v \in R_i^Y$. We consider the cases where both (s_i^Y, s_i^X) and (t_i^Y, t_i^X) are in E (case (a) below), and where at least one of (s_i^Y, s_i^X) and (t_i^Y, t_i^X) is not in E (cases (b) and (c) below). Let $next(v)$ and $prev(v)$ refer to $next(v/Z_i)$ and $prev(v/Z_i)$ respectively.

(a) If $(s_i^X, s_i^Y) \in E$ and $(t_i^X, t_i^Y) \in E$ then Z_i induces a strongly ordered BPG (corollary 4.1) and all shortest paths in Z_i are S-paths (corollary 4.3). If u and v are adjacent to vertices, y' and x', in some shortest path $P(s_i^X, t_i^Y)$, such that $y' \ne t_i^Y$ or $x' \ne s_i^X$, then $dist(u,v) \le dist(s_i^X, t_i^Y)$. Otherwise $prev(t_i^Y) < u$, u is adjacent to t_i^Y, $v < next(s_i^X)$, and v is adjacent to s_i^X. Then $dist(u,v) \le dist(t_i^X, s_i^Y)$ (since by strong ordering, u is adjacent to all neighbors of t_i^X and v is adjacent to all neighbors of s_i^Y). Clearly, the lemma is true for all sets that induce a strongly ordered bipartite permutation graph.

(b) If $(t_i^X, t_i^Y) \notin E$ then $dist(t_i^X, t_i^Y) = 3$ (corollary 4.2). If $next(t_i^Y) \le u$ and $next(t_i^X) \le v$ then $dist(u,v) \le dist(t_i^X, t_i^Y)$ (since $\{u, next(t_i^X), next(t_i^Y), v\}$ is a shortest path between u and v). Assume, in addition, $(s_i^X, s_i^Y) \notin E$, then $prev(s_i^Y) \ge u$ and $prev(s_i^X) \ge v$ implies $dist(u,v) \le dist(s_i^X, s_i^Y)$. Now suppose $u \le prev(s_i^Y)$ and $v \ge next(t_i^X)$. Then by S-ordering every neighbor of t_i^Y is adjacent to $prev(t_i^X)$ and by convexity to v. Likewise every neighbor of s_i^X is adjacent to u. It then follows that $dist(u,v) \le dist(s_i^X, t_i^Y)$. By a similar argument $u \ge next(t_i^Y)$ and $v \le prev(s_i^X)$ implies $dist(u,v) \le dist(s_i^Y, t_i^X)$.

(c) Now suppose $(t_i^X, t_i^Y) \notin E$. If (s_i^X, s_i^Y) is in E then let $x_L = s_i^X$ and $y_L = s_i^Y$. Otherwise $(prev(s_i^X), prev(s_i^Y))$ is in E (by S-ordering), let $x_L = prev(s_i^Y)$ and $y_L = prev(s_i^X)$. Note that if $x_L \ne s_i^X$ then by S-ordering x_L is adjacent to all neighbors of s_i^X. Therefore $dist(x_L, v) \le dist(s_i^X, v)$ for all v in Z_i.

Let G_1 be the (strongly ordered) BPG induced by the set $\{x, y : x_L \le x \le next(t_i^Y), y_L \le y \le next(t_i^X)\}$. Note that if (s_i^Y, s_i^X) is not in E then $prev(s_i^Y) \le next(t_i^Y)$ and $prev(s_i^X) \le next(t_i^X)$ (by S-ordering). Therefore G_1 is not empty.

We already proved that if both u and v are in G_1, or if both are not in G_1 then the claim of the lemma is true (cases (a) and (b) above).

It remains to consider the case where u is in G_1 and v is not in G_1. Suppose that $v > next(t_i^X)$. Therefore v is adjacent to all neighbors of t_i^Y in Z_i (lemma 4.2 and convexity), and u is adjacent to some vertex in $P(x_L, t_i^Y)$. Thus $dist(u, t_i^Y) \leq dist(x_L, t_i^Y) \leq dist(s_i^X, t_i^Y)$. Therefore $dist(u, v) \leq diam(\{s_i^X, s_i^Y, t_i^X, t_i^Y\}/Z_i)$.

Therefore this completes the proof for the case $u \in R_i^X$ and $v \in R_i^Y$.

If $u \in R_i^X$ and $v \in R_i^X$ (or $u \in R_i^Y$ and $v \in R_i^Y$) then consider $P(s_i^X, t_i^X)$ (respectively $P(s_i^Y, t_i^Y)$). Recall that it is an S-path. Since each of u and v is between two vertices in $P^X(s_i^X, t_i^X)$, therefore each is adjacent to some vertex in $P^Y(s_i^X, t_i^X)$ and $dist(u, v) \leq dist(s_i^X, t_i^X)$ (resp. $dist(u, v) \leq dist(s_i^Y, t_i^Y)$). ■

Corollary 4.4 *$diam(Z_i) = diam(\{s_i^X, s_i^Y, t_i^X, t_i^Y\}/Z_i)$.*

Corollary 4.5 *For all x in R_i^X (or y in R_i^Y), $diam(Z_i \bigcup \{x\}) \leq diam(Z_i)$ (respectively $diam(Z_i \bigcup \{y\}) \leq diam(Z_i)$).*

Corollary 4.6 *Let Z_i and Z_j belong to $\mathcal{Z}$, $diam(Z_i) < diam(Z_j)$, and every vertex in Z_j has a neighbor in Z_i, then $diam(Z_i \bigcup Z_j) \leq diam(Z_j)$.*

Proof : For all u and v in $Z_i \bigcup Z_j$, $u, v \in Z_i$ or $u, v \in Z_j$ implies $dist(u, v) \leq diam(Z_j)$, and $u \in Z_i, v \in Z_j$ implies v has a neighbor in Z_i, which, in turn, implies $dist(u, v) \leq diam(Z_i) + 1 \leq diam(Z_j)$. ■

The next three lemmas rely on the above lemma and its corollaries to find maximal sets of a fixed diameter or to break some set into sets of diameters not larger than the original set.

Let W be a subset of $X \bigcup Y$ or be the union of sets in $\mathcal{Z}$. Let s_W^X and s_W^Y be the first vertices in W, and t_W^X and t_W^Y be the last vertices. Let $R(W)$ be the range of W, i.e. the set of vertices between s_W^X, s_W^Y and t_W^X, t_W^Y, inclusive.

Lemma 4.5 (From R-Estar to Estar) *Let W, U, and V be disjoint subsets of $X \bigcup Y$ such that W induces an R-Estar or an L-Estar, every vertex in U is either in $R(W)$ or is greater than $R(W)$ and has a neighbor in W, and every vertex in V is either in $R(W)$ or is smaller than $R(W)$ and has a neighbor in W. Then each of $U \bigcup W$ and $V \bigcup W$ induces a subgraph of diameter three.*

Proof : Suppose W induces an R-Estar. By corollary 4.5 $W' = W \bigcup ((U \bigcup V) \bigcap R(W))$ has diameter three.

Now suppose $U - W' \neq \phi$ and $V - W' \neq \phi$.

Let y_m and x_m be the largest index vertices in U, and let x_U^* and y_U^* be neighbors of y_m and x_m, respectively, in W. Then the edges (y_m, x_U^*) and (s_W^Y, t_W^X) cross, as well as the edges (x_m, y_U^*) and (s_W^X, t_W^Y). By S-ordering one of the edges (y_m, s_W^Y) and (x_U^*, s_W^Y) must be present, as well as one of the edges (x_m, t_W^Y) and (y_U^*, t_W^X). Therefore $U \bigcup W$ is an Estar with center in the set $\{x_U^*, y_U^*, t_W^X, t_W^Y\}$ (by biconvex S-ordering).

Similarly let y_m and x_m be the smallest index vertices in V, and let x_V^* and y_V^* be their neighbors, respectively, in W. Then by convexity x_V^* is adjacent to t_W^Y and y_V^* is adjacent to t_W^X. Therefore, by biconvexity, $V \bigcup W$ is an Estar with center (x_V^*, y_V^*). ■

Lemma 4.6 (Removing/Adding Vertices) *Suppose $diam(Z_i - \{v^*\}) > diam(Z_i)$ for some vertex, v^*, in Z_i. Let T be a set of vertices such that every vertex in T is adjacent to v^*. Then $diam(Z_i \bigcup T) \le diam(Z_i)$.*

Proof : Suppose $v^* \in Z_i^X$. Since $diam(Z_i - \{v^*\}) > diam(Z_i)$, then $diam(Z_i) \ge 2$.

If $diam(Z_i) = 2$ then Z_i is a star with center v^*. If $diam(Z_i) = 3$ then v^* is in every center of Z_i. In both cases it is clear that the lemma is true.

Suppose $diam(Z_i) > 3$. Observe that $dist(t_i, t_j/Z_i \bigcup T) = 2$ for all t_i and t_j in T. Therefore to show that $diam(Z_i \bigcup T) \le diam(Z_i)$, it is adequate to examine the distances between vertices of T and those of Z_i.

If there is a vertex y' in Z_i that has no neighbor other than v^*, then $dist(t, v) \le dist(y', v)$ for all $t \in T$, and $v \in Z_i$.

Otherwise suppose all vertices in Z_i have neighbors in $Z_i - \{v^*\}$.

Let u, v be two vertices in $Z_i - \{v^*\}$ that satisfy $dist(u, v/Z_i - \{v^*\}) > diam(Z_i)$. Let y_a and y_{a+1} be the neighbors of v^* in $P(u, v/Z_i)$, $y_a < y_{a+1}$. Each of y_a and y_{a+1} has neighbors other than v^*. If y_a has a neighbor x larger than v^*, then x is not adjacent to y_{a+1} otherwise $dist(u, v/Z_i - \{v^*\}) \le diam(Z_i)$. Therefore if y_a has a neighbor larger than v^* then y_{a+1} has no neighbor larger than v^*. But now all neighbors of y_{a+1} are smaller than v^* and are adjacent to y_a (by S-ordering), then again $dist(u, v/Z_i - \{v^*\}) \le diam(Z_i)$. Therefore it must be true that all neighbors of y_a, other than v^*, are smaller than v^*, all neighbors of y_{a+1}, other than v^*, are larger than v^*, and $N(y_a) \bigcap N(y_{a+1}) = \{v^*\}$. Therefore, by convexity, all vertices larger than v^* have no neighbors smaller than or equal to y_a, and all vertices smaller than v^* have no neighbors larger than or equal to y_{a+1}.

Therefore $dist(t, v) \le \max(dist(y_{a+1}, v), dist(y_a, v))$ for all $v \in Z_i$ and for all $t \in T$. The explanation for this is as follows. For all t in T, and for all v such that $v < y_{a+1}$ or $v < v^*$, $dist(t, v) \le dist(y_{a+1}, v)$ since there is a shortest path from y_{a+1} to v that passes through v^*. Similarly, for all v such that $v \ge v^*$ or $v \ge y_{a+1}$, $dist(t, v) \le dist(y_a, v)$. ■

Lemma 4.7 *Let Z_i be a set in $\mathcal{Z}$. Let Z_r be a subset of Z_i such that Z_r consists of consecutive vertices in Z_i, $Z_r = \{s_i^X, \ldots, x_{r1}\} \bigcup \{s_i^Y, \ldots, y_{r2}\}$. If each of s_i^X, s_i^Y, x_{r1}, and y_{r2} has a neighbor in Z_r then $diam(Z_r) \le diam(Z_i)$.*

Proof : It is clear that $dist(x_{r1}, y_{r2}/Z_r)$ and $dist(s_i^X, s_i^Y/Z_r)$ are both less than or equal to the respective distances in Z_i. Therefore by corollary 4.4 it suffices to consider the distances: $dist(s_i^X, x_{r1}/Z_r)$, $dist(s_i^Y, x_{r1}/Z_r)$, $dist(s_i^X, y_{r2}/Z_r)$, and $dist(s_i^Y, y_{r2}/Z_r)$. Due to the symmetric nature of biconvex graphs we may safely consider the first two distances only. Now both $P(s_i^X, x_{r1}/Z_i)$ and $P(s_i^Y, x_{r1}/Z_i)$ are S-paths (corollary 4.2). If $P(s_i^X, x_{r1}/Z_i)$ is not contained in Z_r then $prev(x_{r1}/Z_r) = prev(x_{r1}/Z_i)$ is adjacent to some vertex in $P(s_i^X, x_{r1}/Z_i)$ and there is another path through vertices of Z_r of equal length. The same is true for $P(s_i^Y, x_r/Z_r)$. ■

4.4 Parallel Algorithms for Biconvex Graphs

In this section we describe some algorithms that we use in the next chapter. The first algorithm we present constructs a biconvex S-ordering for any biconvex graph. The second algorithm computes the distances between all pairs of vertices in G. Using the above algorithms we show how stars and Estars can be easily recognized.

4.4.1 Ordering of Vertices

Let the adjacency matrix of $G = (X, Y, E)$ have as rows vertices of X and as columns vertices of Y. Then G is biconvex if and only if there is an arrangement of its adjacency matrix that has the consecutive 1's property for both its rows and columns (see [73] for a study of such matrices). In [19] an efficient parallel algorithm is presented to recognize convex graphs and obtain an adjacency matrix that has the consecutive 1's property for the rows. That operation involves only permuting the columns of the adjacency matrix. This implies the algorithm may be easily extended to recognize biconvex graphs and obtain an adjacency matrix that has the consecutive 1's property for both the rows and the columns (by running the algorithm a second time on the transposed matrix). The algorithm uses $\mathcal{O}(n^3)$ processors and $\mathcal{O}(\log^2 n)$ time.

We now present an efficient parallel algorithm to obtain a straight biconvex ordering, given a biconvex ordering. It utilizes $\mathcal{O}(m^2)$ processors, where $m = |E|$, and $\mathcal{O}(\log n)$ time. Its operation relies on the proof of theorem 4.1.

We assume that the input to the algorithm is a biconvex graph $G = (X, Y, E)$ whose vertex set has a biconvex ordering. We assume that its edge set is readily accessible. Such an assumption is reasonable, since given the adjacency matrix we may extract its edge set in $\mathcal{O}(1)$ time using $\mathcal{O}(n^2)$ processors.

Lemma 4.8 (Biconvex S-ordering Algorithm) *Given a biconvex graph $G = (X, Y, E)$, where $X \bigcup Y$ obey a biconvex ordering. $X \bigcup Y$ may be ordered to have a biconvex S-ordering in $\mathcal{O}(\log m)$ time using $\mathcal{O}(m^2)$ processors.*

Proof : We present an algorithm that relies on the proof of theorem 4.1.

Algorithm 4.1 Algorithm S-ORDERING(G)

Input: $G = (X, Y, E)$ *where* G *is a biconvex graph whose vertices have a biconvex ordering,* $X = \{x_1, \ldots, x_{|X|}\}$ and $Y = \{y_1, \ldots, y_{|Y|}\}$.
Output: $G = (X, Y, E)$ *with vertices having a biconvex S-ordering.*

begin

1. $L = T = TT = \phi$
2. *Let* $e_f = (x_{f1}, y_{f2})$ *and* $e_r = (x_{r1}, y_{r2})$ *be two edges that cross,* $x_{f1} < x_{r1}$*, but do not induce an S-ordered biconvex subgraph, i.e. both* (x_{f1}, y_{r2}) *and* (x_{r1}, y_{f2}) *are not in* E*, if no such pair exists then* **exit**
3. $L(f) = e_f$, $L(r) = e_r$
4. **for all** $e_i = (x_{i1}, y_{i2})$ *such that* e_i *crosses* e_f *or* e_r*, and* e_i *has no endpoint in the set* $\{x, y : x_{f1} < x < x_{r1}, y_{f2} < y < y_{r2}\}$ **do in parallel**
   $L(i) = e_i$
   endfor
5. *Sort* L *on* X *endpoints (in parallel)*
6. *Let* $e_s = (x_s, y_s)$ *and* $e_t = (x_t, y_t)$ *be the first and the last edges in* L
7. $L = \phi$
8. **ReverseY:** *Reverse order of* Y *(in parallel)*
9. **for all** $e_i = (x_{i1}, y_{i2})$ *and* $e_j = (x_{j1}, y_{j2})$ *in* E *s.t.* $x_{i1} < x_s, x_{j1} > x_t$ **do in parallel**
   if e_i *crosses* e_j*, and* $(x_{i1}, y_{j2}) \notin E$ *and* $(x_{j1}, y_{i2}) \notin E$ **then** $T(i, j) = 1$
   endfor
   /We use $T(i, j)$ and the next loop to be consistent with our model: CREW/
10. **for all** i **do in parallel**
    $TT(i) = \sum_j T(i, j)$*(in parallel)*
    if $TT(i) > 0$ **then** $L(i) = e_i$
    endfor
11. **if** $L = \phi$ **then exit**
12. *Sort* L *on* X *endpoints (in parallel)*
13. *Let* $e_w = (x_{w1}, y_{w2})$ *be an edge in* L *such that* $x_{w1} < x_s$*,* $w1$ *is the largest such index, let* $e_{w'} = (x_{w3}, y_{w4})$ *be the following edge in* L
14. **ExchangeX:** *Exchange vertices of* $\{x : x_1 \leq x \leq x_{w1}\}$ *and* $\{x : x_{w3} \leq x \leq x_p\}$ *such that* x_i *goes into the* $(p - i + 1)$*th position and* x_{p-i+1} *goes into the* i *th position (in parallel)*

15. **for all** $e_i = (x_{i1}, y_{i2})$ *and* $e_j = (x_{j1}, y_{j2})$ *in* E *s.t.* $y_{i2} < y_s, y_{j2} > y_t$ **do in parallel**
    if e_i *crosses* e_j*, and* $(x_{i1}, y_{j2}) \notin E$ *and* $(x_{j1}, y_{i2}) \notin E$ **then** $T(i,j) = 1$

    endfor

    /We use $T(i,j)$ and the next loop to be consistent with our model: CREW/
16. **for all** i **do in parallel**
    $TT(i) = \sum_j T(i,j)$*(in parallel)*
    if $TT(i) > 0$ **then** $L(i) = e_i$

    endfor
17. **if** $L = \phi$ **then exit**
18. *Sort* L *on* Y *endpoints (in parallel)*
19. *Let* $e_w = (x_{w1}, y_{w2})$ *be the last edge in* L *such that* $y_{w1} < y_s$*,* $w1$ *is the largest such index, let* $e_{w1+1} = (x_{w3}, y_{w4})$ *be the following edge in* L
20. **ExchangeY:** *Exchange vertices of* $\{y : y_1 \leq y \leq y_{w2}\}$ *and* $\{y : y_{w4} \leq y \leq y_p\}$ *such that* y_i *goes into the* $(q-i+1)th$ *position and* y_{q-i+1} *goes into the* i *th position (in parallel)*

end

The correctness of the algorithm follows from theorem 4.1 as it is a direct implementation of the proof of that theorem.

We now show that the algorithm is indeed in NC. The fourth statement and each loop that tests for crossing edges may be executed using $\mathcal{O}(m^2)$ processors: m^2 processors for each pair of edges, one processor and constant time to test whether or not their vertex set induces an S-ordering. Sorting a list of edges needs $\mathcal{O}(m)$ processors and $\mathcal{O}(\log m)$ time [36]. The loop that calculates $TT(i)$, needs $\mathcal{O}(m)$ processors, one for each i, and the sum of $T(i,j)$ may be computed using another $\mathcal{O}(m)$ processors and $\mathcal{O}(\log m)$ time [36].

Therefore, in total, the algorithm executes in $\mathcal{O}(\log m)$ time using $\mathcal{O}(m^2)$ processors. ∎

4.4.2 Distances and Special Graphs

Lemma 4.9 (Computing *prev* **and** *next***)** *For all* v *in* $X \bigcup Y$*,* $prev(v)$ *and* $next(v)$ *can be computed in* $\mathcal{O}(\log\ n)$ *time using* $\mathcal{O}(n)$ *processors. Also for all* Z_i *in* $\mathcal{Z}$ s_i^X*,* s_i^Y*,* t_i^X*, and* t_i^Y *can be computed with the same cost.*

Proof : We use the well known technique of finding the minimum or the maximum of a list of unsorted numbers [36]. ∎

We now show how to compute the distance between all pairs of vertices in a biconvex graph using $\mathcal{O}(n^2)$ processors and $\mathcal{O}(\log n)$ time. It should be mentioned that for general graphs, parallel algorithms that compute the distances between all pairs of vertices use $\mathcal{O}(n^3)$ processors and $\mathcal{O}(\log^2 n)$ time [36].

Lemma 4.10 (Computing Distances) *The distance between all pairs of vertices in G can be computed in parallel in $\mathcal{O}(\log n)$ time using $\mathcal{O}(n^2)$ processors.*

Proof: For all v in $X \bigcup Y$ we assume that $prev(v)$ and $next(v)$ have already been computed then it is easy to see the following. The vertices x and y are adjacent if and only if $prev(y) \leq x \leq next(y)$. The distance between vertices x_i and x_j is two if and only if $prev(x_i) \leq prev(x_j) \leq next(x_i)$ or $prev(x_j) \leq prev(x_i) \leq next(x_j)$. If the path between vertices x_i and y_j is a C-path then $\{x_i, prev(x_i), prev(y_j), y_j\} \in \mathcal{P}(x_i, y_j)$. If $dist(u,v) > 2$ and $P(u,v)$ is an S-path then there is a shortest path $P(u,v) = \{u, next(u), next(next(u)), \ldots, prev(v), v\}$.

Therefore by lemma 4.9, $\mathcal{O}(n)$ processors and $\mathcal{O}(\log n)$ time are needed to compute if two vertices are adjacent, at distance two from each other, or the path between them is a C-path. To compute the distance between vertices u and v otherwise we present the following parallel algorithm.

We assume all distances less than three and all distances of C-paths have already been computed. All other distances are initially set equal to ∞.

Let $next^0(v) = v$, and for $i > 0$ let $next^i(v) = next(next^{i-1}(v))$

The main loop iterates $\log n$ times. After the i th iteration all distances that are less than or equal to $2^i + 1$ are correctly computed. To compute $dist(x,y)$ we use the equation $dist(x,y) = dist(x, foll(x)) + dist(foll(x), y)$, where $foll(x) = next^i(x)$ and $dist(foll(x), y)$ is already computed.

Algorithm 4.2 Algorithm ALL-PAIR-DISTANCES(G)

Input: $G = (X, Y, E)$ *where G is a biconvex graph whose vertices have a biconvex S-ordering,* $X = \{x_1, \ldots, x_{|X|}\}$ and $Y = \{y_1, \ldots, y_{|Y|}\}$.
Output: The distance between u and v $dist(u,v)$ for all $u, v \in X \bigcup Y$.

begin

1. **for all** x_i *in* X, *and* y_j *in* Y **do in parallel**
   $foll(x_i) = next(x_i)$
   $foll(y_i) = next(y_i)$
   endfor
2. $len = 1$.
3. **repeat** $\lceil \log n \rceil$ *times*
   for all u *and* v **do in parallel**
   if $dist(u,v) = \infty$ **then** $dist(u,v) = len + dist(foll(u), v)$
   if $v = foll(u)$ **then** $foll(u) = foll(v)$

/We select processor $(u, foll(u))$ in the above statement to avoid concurrent writes/

$len = len * 2$

endfor

end

The above algorithm computes the distance in $\mathcal{O}(\log n)$ time using $\mathcal{O}(n^2)$ processors. ■

Lemma 4.11 (Recognition of Estars) *Let Z_a be a star, an L-Estar, an R-Estar, or an Estar. Then given s_a^X, s_a^Y, t_a^X, and t_a^Y, Z_a may be recognized in parallel in $\mathcal{O}(\log n)$ time using $\mathcal{O}(n)$ processors.*

Proof : First we compute $prev(v)$ and $next(v)$ for each vertex v in $\{s_a^Y, s_a^X, t_a^Y, t_a^X\}$.

(1) If Z_a is a star then it is sufficient to check that $|Z_a^X| = 1$ (or $|Z_a^Y| = 1$) and $prev(s_a^X) \leq s_a^Y \leq t_a^Y \leq next(s_a^X)$ (resp. $prev(s_a^Y) \leq s_a^X \leq t_a^X \leq next(s_a^Y)$).

(2) If Z_a is an Estar then it is sufficient to check that Z_a contains two stars; an X-star $x^* \bigcup Z_a^Y$ and a Y-star $y^* \bigcup Z_a^X$.

(3) If Z_a is an L-Estar (or an R-Estar) then it is sufficient to check that (t_a^X, t_a^Y) (resp. (s_a^X, s_a^Y)) is an edge, and is a center of Z_a (i.e. every vertex in Z_a is adjacent to one endpoint of that edge). ■

Chapter 5

Partitioning Biconvex Graphs: Structure of Optimal Partition

5.1 Overview and Definitions

In this chapter we make use of results obtained in the last chapter to find an optimal partition, on biconvex graphs for the problem $PVMF$, of special properties that permit efficient parallel and sequential algorithms. The algorithms are presented in the next chapter. The new ordering of the vertices (the biconvex S-ordering) is assumed for a given biconvex graph. Properties such as those proven about diameters of sets, when vertices may be added or removed, will be used frequently. First recall the statement of problem $PVMF$.

> **Problem $PVMF$: Partitioning Vertices to Minimize Monotone Function:** Given $G = (V, E)$, monotone diameter function f, and integers k and B, is there a partition, $\mathcal{Z}$, of V into k or fewer sets, such that $f(\mathcal{Z}) \leq B$?

The definition of monotone diameter functions is given in the introductory chapter (definition 1.1).

In the remainder of this section we introduce some necessary definitions and prove some properties that are true for any optimal partition. In the second section we define a special optimal partition, and present some necessary proofs that will be used in the third section. In the third and last section of this chapter we show that that special optimal partition has some desirable properties that will enable us to design the necessary algorithms to solve $PVMF$ on biconvex graphs.

Some definitions are now due.

Let $G = (X, Y, E)$ be a biconvex graph whose vertices $X = \{x_1, \ldots, x_p\}$ and $Y = \{y_1, \ldots, y_q\}$ have a biconvex S-ordering. Let $n = p + q$ and $m = |E|$.

For any subset $V_1 \subseteq V$, let $< V_1 >$ denote the subgraph induced by vertices of V_1.

Let $\mathcal{Z}$ be a partition for G, $\mathcal{Z} = \{Z_1, \ldots, Z_k\}$. We define the following.

The partition $\mathcal{Z}$ is *optimal* if $f(\mathcal{Z}) \leq B$ and there is no other such partition for G of cardinality smaller than k.

Let Z_a and Z_b be in $\mathcal{Z}$. If $R_a \bigcap R_b \neq \phi$ then Z_a and Z_b are said to *overlap*; otherwise they are *non-overlapping*. Z_a and Z_b are said to *cross* if $R_a^X < R_b^X$ and $R_a^Y > R_b^Y$; otherwise they are *non-crossing*.

A pair of sets (Z_a, Z_b) is a *weak pair* if Z_a and Z_b either overlap or cross; otherwise it is a *strong pair*.

The partition $\mathcal{Z}$ is *overlapping* if it contains a pair of overlapping sets; otherwise it is *non-overlapping*.

The partition $\mathcal{Z}$ is *crossing* if it contains any pair of crossing sets; otherwise it is *non-crossing*.

Figure 5.1 shows a biconvex graph G and a partition $\mathcal{Z}$ for G. In that figure Z_1 and Z_2 overlap, Z_4 and Z_5 cross, and all other set pairs neither cross nor overlap.

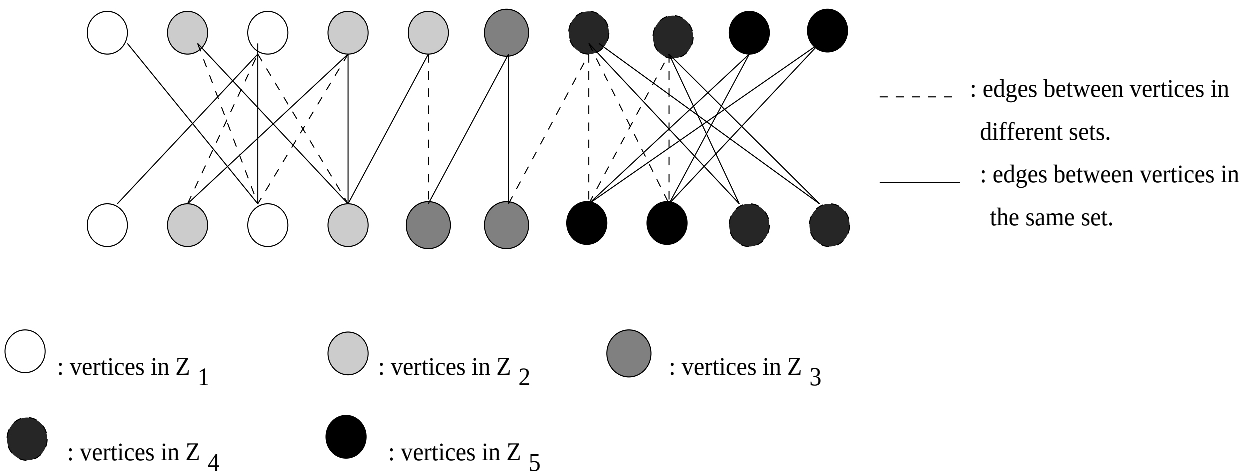

Figure 5.1: A partition and its various sets

It is clear that set pairs can overlap in many ways, and ranges can intersect in many vertices. Optimal partitions that generally allow many overlap patterns without restrictions, particularly on the number of vertices in the intersection of ranges, do not seem to readily permit efficient parallel algorithms. We would like to have an optimal partition where overlaps and crossings are limited to special recognizable cases.

In the rest of this chapter we show that there is a special optimal partition, where overlaps and crossings always occur at the 'end' of the graph and between sets of bounded diameter. We prove this by studying all patterns of overlap and crossing between set pairs, eliminate some and reduce others to special cases.

Figure 5.2 shows all possible overlap and crossing patterns between any set pair (Z_i, Z_j) in an optimal partition. In that figure the roles of Z_i and Z_j may be interchanged in any pattern. We group them into five groups. For any set pair (Z_i, Z_j), we say the pair (Z_i, Z_j) forms pattern α if their overlap pattern is one of those in group α. Table 5.1 formally defines overlaps and crossings.

Pattern	*Description*
Pattern 1	$R_j^Y \subset R_i^Y$ and $R_i^X \not\subset R_j^X$ or $R_j^X \subset R_i^X$ and $R_i^Y \not\subset R_j^Y$
Pattern 2	$(R_j - R_i)^X > R_i^X$ and $(R_j - R_i)^Y > R_i^Y$
Pattern 3	$R_j^Y \subset R_i^Y$ and $R_i^X \subset R_j^X$
Pattern 4	$(R_j - R_i)^X > R_i^X$ and $(R_j - R_i)^Y < R_i^Y$, and $((R_i \bigcap R_j)^Y \neq \phi$ or $(R_i \bigcap R_j)^X \neq \phi)$
Pattern 5 : Crossing Sets	$R_j^Y < R_i^Y$ and $R_j^X > R_i^X$

Table 5.1 : Crossing and Overlap Patterns for a pair (Z_i, Z_j)

We now show that this table is, indeed, exhaustive. For any pair of sets, (Z_i, Z_j) in an optimal partition $\mathcal{Z}$, if $(R_i \bigcap R_j)^Y = \phi$ and $(R_i \bigcap R_j)^X = \phi$, then Z_i and Z_j do not overlap but they may cross **(pattern 5)**.

Now consider any pair of sets (Z_i, Z_j), in an optimal partition $\mathcal{Z}$, that overlap (i.e. at least one of $(R_i \bigcap R_j)^Y$ and $(R_i \bigcap R_j)^X$ is not empty). We have one of the following cases. We will omit cases implied by the horizontal symmetry in biconvex graphs, that is cases obtained by interchanging the superscripts X and Y. Also the subscripts i and j may be interchanged.

(a) $R_j^Y \subset R_i^Y$, or $R_j^X \subset R_i^X$, or $R_i^Y \subset R_j^Y$, or $R_i^X \subset R_j^X$: Suppose $R_j^Y \subset R_i^Y$ (the other cases follow similarly), then it cannot be true that $R_j^X \subset R_i^X$ (otherwise the partition is not optimal by corollary 4.5). One of the following is true.

(a.1) $R_i^X \subset R_j^X$: giving **pattern 3**.

(a.2) $R_i^X \not\subset R_j^X$: giving **pattern 1**.

(b) $(R_j - R_i)^X > R_i^X$ or $(R_j - R_i)^X < R_i^X$: Suppose $(R_j - R_i)^X > R_i^X$ (the other case follows similarly), then one of the following is true.

(b.1) $(R_j - R_i)^Y < R_i^Y$: giving **pattern 4**.

(b.2) $(R_j - R_i)^Y > R_i^Y$: giving **pattern 2**.

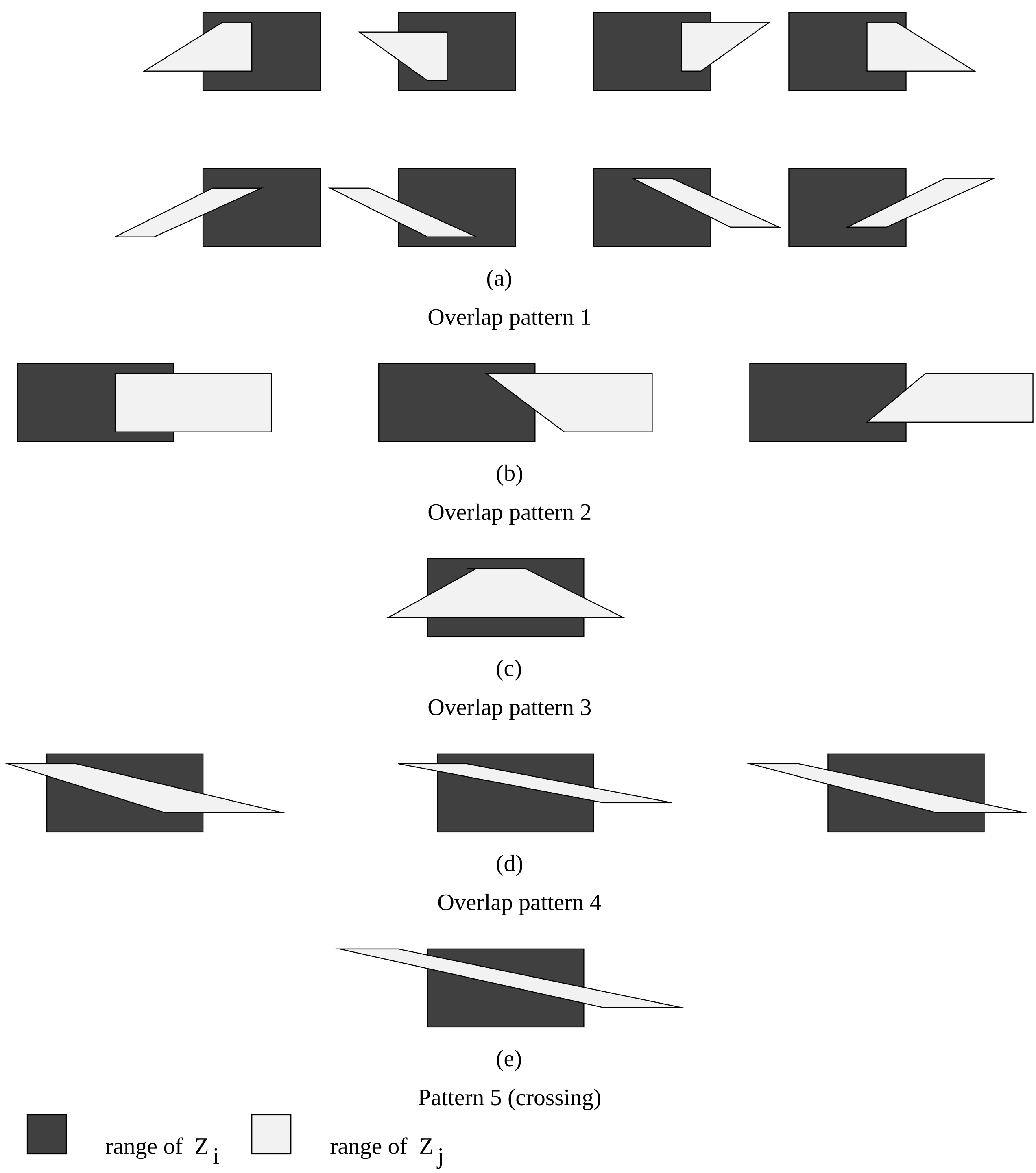

Figure 5.2: Crossing and overlap patterns for a weak pair (Z_i, Z_j) in an optimal partition

Before we end this section we have a few proofs that are true for <u>any</u> optimal partition. The first few proofs are propositions that are direct consequences of lemmas in chapter

4, nevertheless we use them often enough to warrant mentioning them separately. These arguments now follow.

Proposition 5.1 *Let $< W >$ be a subgraph of G that induces an Estar. Then W may be partitioned into two stars.*

Proof : Let (x^*, y^*) be a center of $< W >$ then $\{Z_1, Z_2\}$ is a partition that satisfies the proposition, where $Z_1 = \{x^*\} \bigcup W^Y - \{y^*\}$ and $Z_2 = \{y^*\} \bigcup W^X - \{x^*\}$. ■

Let $\mathcal{Z}$ be an optimal partition for G, $\mathcal{Z} = \{Z_1, .., Z_a, ..Z_b, ..Z_k\}$.

Proposition 5.2 *Given sets Z_a and Z_b, $b \neq a$. If $Z_a = \{y\}$, x is a neighbor of y, and $x \in Z_b$, then $diam(Z_b - \{x\}) \leq diam(Z_b)$.*

Proof : Otherwise we will have a smaller partition by lemma 4.6. ■

Proposition 5.3 *Given set Z_a. If $y \in R_a$ then y has a neighbor in Z_a.*

Proof : By convexity since Z_a is connected. ■

Proposition 5.4 *Given sets Z_a and Z_b. It cannot be true that $R_b^Y \subset R_a^Y$ or every vertex in Z_b^Y is adjacent to all neighbors, in Z_a, of a vertex in Z_a^Y, and $R_b^X \subset R_a^X$ or every vertex in Z_b^X is adjacent to all neighbors, in Z_a, of a vertex in Z_a^X.*

Proof : If $R_b^Y \subset R_a^Y$ then by corollary 4.5 we have $diam(Z_a \bigcup Z_b^Y) = diam(Z_a)$. If every vertex in Z_b^Y is adjacent to all the neighbors of a vertex in Z_a, then for every vertex y in Z_b^Y, there is a vertex y' in Z_a^Y such that $dist(y, v/Z_a \bigcup Z_b) \leq dist(y', v/Z_a)$, for all vertices v in Z_a, and therefore $diam(Z_a \bigcup Z_b^Y) = diam(Z_a)$.

Therefore if $R_b^Y \subset R_a^Y$ or every vertex in Z_b^Y is adjacent to all neighbors of a vertex in Z_a^Y, and $R_b^X \subset R_a^X$ or every vertex in Z_b^X is adjacent to all neighbors of a vertex in Z_a^X, then there is a smaller partition, which is a contradiction. ■

We now prove some other properties satisfied by any optimal partition; any two sets that form pattern 3 or pattern 4 (figures 5.2 c and 5.2 d) have equal diameters (lemma 5.1), any two sets that cross (figure 5.2 e) induce a diameter two subgraph, an R-Estar or an L-Estar (lemma 5.2), and if two diameter two sets induce an R-Estar (or an L-Estar) then they contain x_1 and y_1 (respectively x_p and y_q) (lemma 5.3).

Let $\mathcal{Z}$ be an optimal partition.

Lemma 5.1 (Patterns 3 and 4: First lemma) *Let Z_a and Z_b be two sets that overlap in pattern 3 or in pattern 4, then $diam(Z_a) = diam(Z_b)$, and $diam(Z_a \bigcup Z_b) = diam(Z_b) + 1$. Furthermore if $diam(Z_b) = 3$ then (Z_a, Z_b) forms pattern 4.*

Proof: It is clear that both Z_a and Z_b have diameters greater than one (by the definitions of the overlap patterns). Suppose $diam(Z_a) \geq diam(Z_b)$, $R_a^X \subset R_b^X$ if the pair forms pattern 3, $R_b^X \bigcap R_a^X \neq \phi$ and $s_a^X < s_b^X$ if the pair forms pattern 4.

First we prove that both sets have equal diameter and their union has a diameter that equals one plus the diameter of either set.

In pattern 3 every vertex in Z_a^X is in R_b, therefore $diam(Z_b \bigcup Z_a^X) \leq diam(Z_b)$ (see corollary 4.5). Every vertex in Z_a^Y has some neighbor in $Z_a^X \bigcup Z_b$ (see figure 5.2 c). Therefore for all $v \in Z_a^Y$ and $u \in Z_b$, $dist(u,v) \leq diam(Z_b \bigcup Z_a^X) + 1 \leq diam(Z_b) + 1$. Therefore if $diam(Z_b) < diam(Z_a)$ then $diam(Z_a \bigcup Z_b) \leq diam(Z_a)$, and $\mathcal{Z}$ is not minimal, which is a contradiction.

Now let Z_a and Z_b form pattern 4. Let all distances refer to distances in $Z_a \bigcup Z_b$. The distance $dist(s_a^X, s_b^Y)$ is less than or equal to three (by S-ordering since the edges $(s_a^X, prev(s_a^X/Z_a))$ and $(s_b^Y, prev(s_b^Y/Z_b))$ cross). Similarly the distance $dist(t_b^X, t_a^Y)$ is not larger than three. The distance $dist(s_a^X, t_b^X)$ is equal to two if $prev(s_a^X/Z_a) > t_b^Y$ (by S-ordering since the edges $(s_a^X, prev(s_a^X/Z_a))$ and $(t_b^X, next(t_b^X/Z_b))$ will cross), or is less than or equal to $diam(Z_b) + 1$ otherwise (by corollary 4.5 since $prev(s_a^X/Z_a)$ will be in R_b). Similarly, the distance $dist(s_b^Y, t_a^Y)$ is less than or equal to $diam(Z_b) + 1$. Therefore $diam(Z_a \bigcup Z_b) \leq \max(diam(Z_a), diam(Z_b) + 1)$ (see corollaries 4.4 and 4.5). Therefore if $diam(Z_a) > diam(Z_b)$ then $diam(Z_a \bigcup Z_b) \leq diam(Z_a)$ and $\mathcal{Z}$ is not minimal, which is a contradiction.

Therefore it must be true that $diam(Z_a) = diam(Z_b)$, and $diam(Z_a \bigcup Z_b) = diam(Z_b)+1$.

Now suppose $diam(Z_a) = diam(Z_b) = 3$ and the pair forms pattern 3. Then, by biconvexity, $Z_a \bigcup Z_b$ is an Estar with center (x_a^*, y_b^*), where x_a^* is in a center of Z_a, and y_b^* is in a center of Z_b, a contradiction to the minimality of $\mathcal{Z}$. ∎

Lemma 5.2 (Crossing Partitions: First lemma) *If Z_a and Z_b are two sets in $\mathcal{Z}$ such that $\{Z_a, Z_b\}$ is a crossing subpartition (pattern 5), then the following is true.*
(a) $diam(Z_a), diam(Z_b) \leq 2$.
(b) $Z_a \bigcup Z_b$ induces a subgraph of diameter two, an R-Estar or an L-Estar.

Proof : Let Z_a, Z_b be two sets in $\mathcal{Z}$ such that the subpartition of $Z_a \bigcup Z_b$ is a crossing one. Suppose $diam(Z_a) \leq diam(Z_b)$.

We may assume that $R_b^Y < R_a^Y$ and $R_b^X > R_a^X$ (see figure 5.1 e). Let all distances refer to distances in $Z_a \bigcup Z_b$. Every edge in Z_b crosses every edge in Z_a. Therefore, by S-ordering of vertices $dist(s_a^X, t_b^X) = dist(s_b^Y, t_a^Y) = 2$. Consequently, by biconvexity, there are vertices x^* and y^* in $Z_a \bigcup Z_b$ such that x^* is adjacent to all vertices in $(Z_a \bigcup Z_b)^Y$ and y^* is adjacent to all vertices in $(Z_a \bigcup Z_b)^X$. It follows that $Z_a \bigcup Z_b$ induces a subgraph of diameter not larger than three. Since $\mathcal{Z}$ is minimal then it must be true that both diameters are less than or equal to two. This proves part (a).

Since both sets have diameters smaller than three, then the edges (s_a^X, t_a^Y) and (s_b^Y, t_b^X) must exist. The edges (s_a^X, t_a^Y) and (s_b^Y, t_b^X) cross. Therefore $Z_a \bigcup Z_b$ induces a subgraph of diameter two, an L-Estar or an R-Estar (by biconvex S-ordering of the vertices). This completes the proof for part (b) and the lemma. ∎

For any subset $\mathcal{W}$ of $\mathcal{Z}$, let s_W^X and s_W^Y be the smallest index vertices in any set in $\mathcal{W}$, and let t_W^X and t_W^Y be the largest index vertices in any set in $\mathcal{W}$. The range of $\mathcal{W}$, $R(\mathcal{W})$, is the set of all vertices between s_W^X, s_W^Y and t_W^X, t_W^Y inclusive.

Lemma 5.3 (Cardinality two Estars at the End of the Graph) *Let $\mathcal{W}$ be a subset of $\mathcal{Z}$ such that $|\mathcal{W}| = 2$, $\mathcal{W}$ consists of two diameter two sets, and $< \mathcal{W} >$ is an R-Estar (or an L-Estar). Then each of x_1 and y_1 (respectively x_p and y_q) is in some set in $\mathcal{W}$.*

Proof : Suppose $< \mathcal{W} >$ is an R-Estar then (t_W^X, t_W^Y) is a center of $\mathcal{W}$. Assume $y_1 \in Z_b$ and $Z_b \notin \mathcal{W}$. Since $< \mathcal{W} >$ is an R-Estar then y_1 has no neighbor smaller than or equal to s_W^X (otherwise by lemma 4.2 and convexity s_W^X is adjacent to s_W^Y and $\mathcal{W}$ has diameter two and is thus not an Estar). Therefore every edge incident with y_1 crosses (s_W^X, t_W^Y), t_W^Y is adjacent to all neighbors of y_1 (by biconvex S-ordering and the fact that y_1 is not adjacent to s_W^X), and hence all neighbors of y_1 are adjacent to all vertices y in $R^Y(\mathcal{W})$ (by convexity).

If $|Z_b| = 1$ then let $prev(y_1) \in Z_c$. $prev(y_1)$ is adjacent to all vertices in $R^Y(\mathcal{W})$, and $diam(Z_c - \{prev(y_1)\}) \leq diam(Z_c)$ (by proposition 5.2). Let $Z_c = Z_c - \{prev(y_1)\}$ and $Z_b = \{y_1, prev(y_1)\}$.

Let $Z_b' = Z_b \bigcup \{y : y \in \mathcal{W}, y \neq t_W^Y\}$. Let W be the set of vertices in any set in $\mathcal{W}$. Let $W' = W - Z_b'$. $< W' >$ induces a Y-star. $diam(Z_b') = 2$ or if $diam(Z_b) > 2$ then $diam(Z_b') = diam(Z_b)$. Let $\mathcal{Z}' = \mathcal{Z} - \mathcal{W} - \{Z_b\} \bigcup \{W', Z_b'\}$. Therefore we have a partition, $\mathcal{Z}'$, of smaller cardinality and $f(\mathcal{Z}') \leq f(\mathcal{Z})$; which contradicts the optimality of $\mathcal{Z}$. ∎

5.2 Minimal Overlap Strong Optimal Partition

In this section we give definitions of special optimal partitions with desirable properties. We prove some properties that are true in any optimal partition that has the minimum number of weak set pairs (among all optimal partitions for a certain graph). We will call this partition a *strong* partition. We then define what we call *a minimal overlap strong optimal partition*, that is a strong optimal partition with additional requirements on weak pairs.

> **Definition 5.1:** $\mathcal{Z}$ is a *strong* optimal partition for G, if $\mathcal{Z}$ is an optimal partition for G, and $\mathcal{Z}$ has the minimum number of weak set pairs among all optimal partitions for G.

Let $\mathcal{Z}$ be a strong optimal partition. For any weak pair, (Z_a, Z_b) in $\mathcal{Z}$, there cannot exist sets Z_a' and Z_b' such that $Z_a' \bigcup Z_b' = Z_a \bigcup Z_b$, the pair (Z_a', Z_b') is strong, and the partition $\mathcal{Z}' = \mathcal{Z} - \{Z_a, Z_b\} \bigcup \{Z_a', Z_b'\}$ has fewer weak pairs. The following theorem then applies.

Theorem 5.1 *If $\mathcal{Z}$ is a strong optimal partition, then for every weak pair (Z_a, Z_b) in $\mathcal{Z}$, it <u>cannot</u> be true that there are vertices x_L, y_L, x_R, and y_R in $Z_a \bigcup Z_b$ such that the following is true.*
(a) At least one of x_R and x_L is in $R_a \bigcap R_b$, and at least one of y_R and y_L is in $R_a \bigcap R_b$.
(b) x_L immediately precedes x_R in $Z_a \bigcup Z_b$, and y_L immediately precedes y_R in $Z_a \bigcup Z_b$.
(c) The sets $Z_a' = \{x, y : x, y \in Z_a \bigcup Z_b, x \leq x_L, y \leq y_L\}$ and $Z_b' = \{x, y : x, y \in Z_a \bigcup Z_b, x \geq x_R, y \geq y_R\}$ are such that one has diameter not larger than that of Z_a and the other has diameter not larger than that of Z_b.

Proof: Suppose not. Let $\mathcal{Z}$ be a strong optimal partition. Let (Z_a, Z_b) be a weak pair in $\mathcal{Z}$ such that Z_a overlaps Z_b. If no such pair exists then the theorem is trivially true.

Let vertices x_L, y_L, x_R, and y_R in $Z_a \bigcup Z_b$ satisfy conditions (a) and (b) of the theorem. Let $Z_a' = \{x, y : x, y \in Z_a \bigcup Z_b, x \leq x_L, y \leq y_L\}$ and $Z_b' = \{x, y : x, y \in Z_a \bigcup Z_b, x \geq x_R, y \geq y_R\}$, and suppose $diam(Z_a')$ is not larger than that of Z_a and $diam(Z_b')$ is not larger than that of Z_b. $Z_a' \bigcup Z_b' = Z_a \bigcup Z_b$, Z_a' does not overlap Z_b', and $Z_a' < Z_b'$.

Consider now the partition $\mathcal{Z}' = \mathcal{Z} - \{Z_a, Z_b\} \bigcup \{Z_a', Z_b'\}$. $\mathcal{Z}'$ is optimal since $|\mathcal{Z}'| = |\mathcal{Z}|$ and $f(\mathcal{Z}') \leq f(\mathcal{Z})$.

Suppose $\mathcal{Z}'$ has more weak set pairs. Any weak pair in $\mathcal{Z}'$ that does not include any of Z_a' and Z_b', corresponds to a weak pair in $\mathcal{Z}$. Any set that crosses both Z_a' and Z_b', crosses both Z_a and Z_b. Any set that overlaps Z_a' or Z_b' overlaps at least one of Z_a or Z_b. Therefore if $\mathcal{Z}'$ had more weak pairs than $\mathcal{Z}$, then there would a set Z_c such that both pairs (Z_c, Z_a') and (Z_c, Z_b') are weak, but only one of (Z_c, Z_a) and (Z_c, Z_b) is weak.

Let Z_c be a set that overlaps or crosses exactly one of Z_a and Z_b.

Suppose x_L is in $R_a \bigcap R_b$. Then x_L is not in R_c, and at least one of y_L or y_R, whichever is in $R_a \bigcap R_b$, is not in R_c either.

Let R_a' be the range of Z_a' and R_b' be that of Z_b'.

If R_c intersects $R_a'^Y$ and $R_b'^X$ only, or R_c intersects $R_a'^X$ and $R_b'^Y$ only then Z_c overlaps or crosses each of Z_a and Z_b. We elaborate on this. Suppose R_c intersects $R_a'^Y$ and $R_b'^X$ only, and Z_c neither crosses nor overlaps Z_b. Then either $R_b < R_c$ or $R_b > R_c$. Since x_L is in R_b and is not in R_c, therefore $R_b < R_c$. Let $y = y_L$ or $y = y_R$, depending on whether $y_L \in R_a \bigcap R_b$ or $y_R \in R_a \bigcap R_b$, respectively. Therefore $y > R_c^Y$. But now y cannot be in R_b, a contradiction.

Therefore R_c intersects $R_a'^Y$ and $R_b'^Y$, or R_c intersects $R_a'^X$ and $R_b'^X$. Then either both of x_L and x_R, or both of y_L, and y_R are in R_c. By our choice of these vertices, it follows that Z_c overlaps both Z_a and Z_b.

Therefore $\mathcal{Z}'$ has more strong pairs, a contradiction to our choice of $\mathcal{Z}$. ■

Lemma 5.4 (Crossing Partitions: Second lemma) *Let $\mathcal{Z}$ be a strong optimal partition that contains a pair of sets, (Z_a, Z_b) that cross, then the following is true.*
(a) $Z_a \bigcup Z_b$ induces an R-Estar or an L-Estar.
(b) $diam(Z_a) = diam(Z_b) \leq 2$.
(c) If some set Z_c crosses one or both of Z_a or Z_b then all three diameters are equal to one.

Proof : Let $\mathcal{Z}$ be a strong optimal partition. Let Z_a, Z_b be two sets in $\mathcal{Z}$ that cross. By lemma 5.2 both diameters are smaller than three, and $Z_a \bigcup Z_b$ induces a subgraph of diameter two, an L-Estar or an R-Estar.

We now show that if (a) is not satisfied then $\mathcal{Z}$ is not strong. Suppose Z_a crosses Z_b, $Z_b^Y < Z_a^Y$, and the subgraph $< Z_a \bigcup Z_b >$ has diameter two. Then it must be true that both sets have diameter one. Let $Z_a' = \{s_b^Y, s_a^X\}$ and $Z_b' = \{t_a^Y, t_b^X\}$. The new pair is strong, and both sets have diameter one. Therefore $\mathcal{Z}' = \mathcal{Z} - \{Z_a, Z_b\} \bigcup \{Z_a', Z_b'\}$ is optimal. We now show that $\mathcal{Z}'$ has fewer weak pairs, hence a contradiction to our choice of $\mathcal{Z}$. Any set that crosses both Z_a' and Z_b' crosses both Z_a and Z_b. Suppose Z_c crosses Z_a' only, for instance. Then $Z_c^X < Z_a'^X$ implies that Z_c crosses Z_b, and $Z_c^X > Z_a'^X$ implies that Z_c crosses Z_a. Any set that overlaps Z_a', for instance, contains s_b^Y or s_a^X and therefore overlaps Z_b or Z_a. Suppose Z_c overlaps both Z_a' and Z_b' but overlaps Z_a only. Then both s_a^X and t_a^Y are in R_c. But now Z_b crosses Z_c. Therefore the number of strong pairs in $\mathcal{Z}'$ is larger than that in $\mathcal{Z}$. This completes the proof for part (a).

We now prove part (b). We may assume that $R_b^Y < R_a^Y$, and that $< Z_a \bigcup Z_b >$ is an L-Estar. Suppose $diam(Z_a) = 2$ and $diam(Z_b) = 1$. s_b^Y is adjacent to all vertices in Z_a^X (since $< Z_a \bigcup Z_b >$ is an L-Estar). Let $Z_a' = Z_a \bigcup \{s_b^Y\}$ and $Z_b' = \{s_b^X\}$ (see figure 5.3).

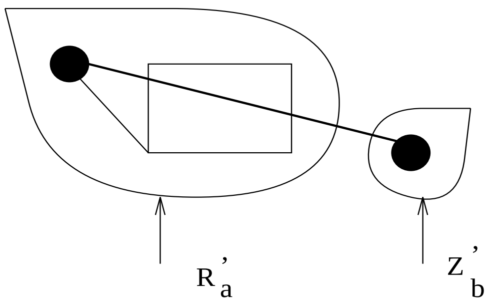

Figure 5.3: Z_a **crosses** Z_b, $|Z_b| = 2$

No set overlaps Z_b' (otherwise $\mathcal{Z}'$ is not optimal). Any set that crosses Z_a', crosses Z_a. Suppose Z_c overlaps Z_a' but does not overlap Z_a or Z_b, then either $R_b^Y < R_c^Y < R_a^Y$ and $R_c^X < R_a^X$ in which case Z_c crosses Z_b, or $R_b^Y < R_c^Y < R_a^Y$ and $R_c^X > R_a^X$ in which case Z_c crosses Z_a.

Therefore $\mathcal{Z}'$ has more strong pairs, a contradiction to our choice of $\mathcal{Z}$.

We now prove part (c). Suppose $diam(Z_a) = diam(Z_b) = 2$, and there is a set Z_c that crosses one or both of Z_a and Z_b. If Z_c crosses both then either $Z_c^Y < Z_b^Y$ and $Z_c^X > Z_b^X$ in which case Z_b is a subset of $R(Z_c \bigcup Z_a)$, or $Z_c^Y > Z_a^Y$ and $Z_c^X < Z_a^X$ in which case Z_a is a subset of $R(Z_c \bigcup Z_b)$. Therefore $Z_a \bigcup Z_b \bigcup Z_c$ is an L-Estar or an R-Estar by lemma 5.2 and corollary 4.5, and by proposition 5.1 we have a contradiction to the minimality of $\mathcal{Z}$. Otherwise Z_c crosses one only of Z_a or Z_b. In this case either $Z_b^Y < Z_c^Y < Z_a^Y$ and $Z_c^X < Z_a^X$ or $Z_c^X > Z_b^X$, or $Z_a^X < Z_c^X < Z_b^X$ and $Z_c^Y > Z_a^Y$ or $Z_c^Y < Z_b^Y$. In either case, since $Z_a \bigcup Z_b$ is an L-Estar or an R-Estar, by lemma 4.5 and proposition 5.1 we have a smaller partition, which is a contradiction to our choice of $\mathcal{Z}$. ∎

We now show that there is a special optimal partition which we call *a minimal overlap strong optimal partition*, that by definition does not include, or may be made not to include, some of the overlap patterns listed in table 5.1, and those patterns that it includes are limited to special cases (as will be shown in the next section).

We will shortly give a definition showing how set pairs in the so called minimal overlap optimal partition should overlap. However we first give an intuitive explanation for that definition.

Our ultimate goal is to find an optimal partition where the number of vertices that are in more than one range is bounded above by an integer, thus facilitating our job of developing an efficient parallel algorithm.

In a minimal overlap optimal partition, $\mathcal{Z}$, we want the ranges of set pairs to overlap in as few vertices as possible. To exemplify, if a set pair (Z_a, Z_b), in $\mathcal{Z}$, overlaps, then we want $|R_a \bigcap R_b|$ to be as small as possible; in the sense that removing any vertices from $(Z_a \bigcap R_b)$ and adding them to Z_b would increase the diameter of Z_a, and removing any vertices from $(Z_b \bigcap R_a)$ and adding them to Z_a would increase the diameter of Z_b. We say that such a pair *minimally overlap*. Note that by corollary 4.5 adding any vertex in $R_a \bigcap R_b$ to Z_a (or to Z_b) does not increase the diameter of Z_a (resp. of Z_b). Thus if (Z_a, Z_b) do <u>not</u> minimally overlap, then there is a set in $R_a \bigcap R_b$ that may be moved from one set to the other without affecting the optimality of the partition, and such a movement would cause the cardinality of $R_a \bigcap R_b$ to decrease. By the definition of ranges, the cardinality of $R = R_a \bigcap R_b$ will decrease only if at least one of its end vertices is removed (i.e. one of the smallest index vertices or the largest index vertices). Since the end vertices of R are in the set $\{s_a^Y, s_a^X, t_a^Y, t_a^X\} \bigcup \{s_b^Y, s_b^X, t_b^Y, t_b^X\}$, therefore we can move at least one of the end vertices of Z_a or Z_b or both, from one set to the other.

Now we are ready for the necessary definitions.

Definition 5.2: A pair of sets (Z_a, Z_b) in $\mathcal{Z}$ is said to *minimally overlap* if $R_a \bigcap R_b \neq \phi$ implies there do <u>not</u> exist sets S and T, where S is a subset of

$Z_a \bigcap R_b$, T is a subset of $Z_b \bigcap R_a$, $S \neq \phi$ or $T \neq \phi$, and the following is true.

(1.a) $S = S_L^X \bigcup S_L^Y \bigcup S_R^X \bigcup S_R^Y$, the sets S_L^X, S_L^Y, S_R^X, and S_R^Y are disjoint, one or more may be empty, each of S_L^X and S_R^X is a subset of Z_a^X, and each of S_L^Y and S_R^Y is a subset of Z_a^Y.

(1.b) $T = T_L^X \bigcup T_L^Y \bigcup T_R^X \bigcup T_R^Y$, the sets T_L^X, T_L^Y, T_R^X, and T_R^Y are disjoint, one or more may be empty, each of T_L^X and T_R^X is a subset of Z_b^X, and each of T_L^Y and T_R^Y is a subset of Z_b^Y.

(2.a) $s_a^X \in S_L^X$, $s_a^Y \in S_L^Y$, $t_a^X \in S_R^X$, and $t_a^Y \in S_R^Y$, provided $S_L^X \neq \phi$, $S_L^Y \neq \phi$, $S_R^X \neq \phi$, $S_R^Y \neq \phi$, respectively.

(2.b) $s_b^X \in T_L^X$, $s_b^Y \in T_L^Y$, $t_b^X \in T_R^X$, and $t_b^Y \in T_R^Y$, provided $T_L^X \neq \phi$, $T_L^Y \neq \phi$, $T_R^X \neq \phi$, $T_R^Y \neq \phi$, respectively.

(3.a) Each of S_L^X, S_L^Y, S_R^X, and S_R^Y is either empty or consists of consecutive vertices in Z_a.

(3.b) Each of T_L^X, T_L^Y, T_R^X, and T_R^Y is either empty or consists of consecutive vertices in Z_b.

(4) $diam((Z_a \bigcup T) - S) \leq diam(Z_a)$ and $diam((Z_b \bigcup S) - T) \leq diam(Z_b)$.

Note that given a pair (Z_a, Z_b) that does not minimally overlap, it will not be true that all four sets S_L^X, S_L^Y, S_R^X, and S_R^Y are non empty, or all four sets T_L^X, T_L^Y, T_R^X, and T_R^Y are non empty (see figure 5.2).

Definition 5.3: $\mathcal{Z}$ is a *minimal overlap partition*, if for every pair of sets (Z_a, Z_b) in $\mathcal{Z}$, (Z_a, Z_b) minimally overlap.

For example figure 5.4 a shows an optimal partition that is <u>not</u> a minimal overlap partition, and figure 5.4 b shows a minimal overlap optimal partition for the same graph. All this is with respect to the particular biconvex S-ordering.

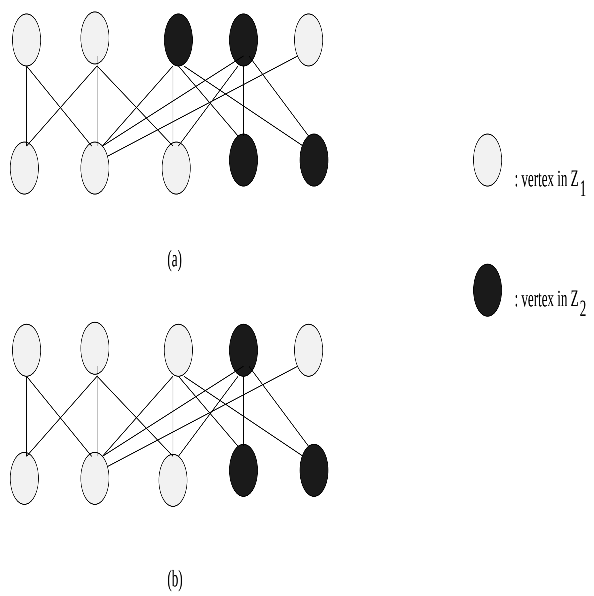

Figure 5.4: (a) An optimal partition for a graph G, $B = 3$, $\bullet = \max$ **(b) A minimal overlap optimal partition for** G

We now show that any biconvex graph has a minimal overlap strong optimal partition.

Theorem 5.2 *Given a biconvex graph G, G has a minimal overlap strong optimal partition.*

Proof: If not then suppose $\mathcal{Z}$ is a strong optimal partition for G that contains a pair of sets (Z_a, Z_b) that do not minimally overlap. The proof proceeds constructively as follows.

Let $\mathcal{Z}^0 = \mathcal{Z}$, $Z_i^0 = Z_i$ for all i, and $it = 1$.

begin

While there is a pair of sets, (Z_a^{it-1}, Z_b^{it-1}), in $\mathcal{Z}^{it-1}$, that do not minimally overlap, do the following.

1. Find sets S and T, such that $S \subseteq Z_a^{it-1} \bigcap R_b^{it-1}$ and $T \subseteq Z_b^{it-1} \bigcap R_a^{it-1}$, and each of S and T contains at least one of the end vertices of Z_a^{it-1} and Z_b^{it-1}, and consists of consecutive vertices in Z_a^{it-1} and Z_b^{it-1}, respectively, in a manner consistent with definition 5.2, and $diam((Z_a^{it-1} - S) \bigcup T) \leq diam(Z_a^{it-1})$, and $diam((Z_b^{it-1} - T) \bigcup S) \leq diam(Z_b^{it-1})$. At least one of S or T exists by the definition of minimally overlapping set pairs (definition 5.2). Let S and T be maximal such sets.

2. Let $Z_a^{it} = (Z_a^{it-1} - S) \bigcup T$, $Z_b^{it} = (Z_b^{it-1} \bigcup S) - T$, and $\mathcal{Z}^{it} = \mathcal{Z}^{it-1} - \{Z_a^{it-1}, Z_b^{it-1}\} \bigcup \{Z_a^{it}, Z_b^{it}\}$.

3. Increment it.

enddo.

end

Observe the following:

(i) Each partition $\mathcal{Z}^{it}$ is an optimal partition, by our initial choice of $\mathcal{Z}^0$, and by our choice of the sets S and T.

(ii) Any strong pair will not be chosen in the above loop, therefore by our choice of $\mathcal{Z}^0$, $\mathcal{Z}^{it}$ is a strong optimal partition.

(iii) $|R_a^{it+1} \bigcap R_b^{it+1}| < |R_a^{it} \bigcap R_b^{it}|$, for any pair (Z_a, Z_b) chosen in the above loop by our choice of sets S and T, for all other sets Z_i^{it} and Z_j^{it}, $(R_i^{it+1} \bigcap R_j^{it+1}) = (R_i^{it} \bigcap R_j^{it})$ if both i and j are not in $\{a, b\}$, and if exactly one of i or j is in $\{a, b\}$ then $|R_i^{it+1} \bigcap R_j^{it+1}| \leq |R_i^{it} \bigcap R_j^{it}|$.

If the above procedure halts at some $it = f$, then it is clear that $\mathcal{Z}^f$ is a minimal overlap strong optimal partition.

We now show that the procedure does indeed halt. Let W^{it} be a set such that v_i^j is in W^{it} if and only if v_i is in the intersection of j or more ranges of sets in $\mathcal{Z}^{it}$, i.e. there is at least one copy of each vertex of G in W^{it}, and if v_i is in l ranges of sets in $\mathcal{Z}^{it}$, then there are l copies of v_i in W^{it}: $v_i^1, v_i^2, \ldots, v_i^l$. Then $|W^{it+1}| < |W^{it}|$ (by observation (iii) above). Therefore the above procedure executes a number of times not larger than $|W^0| \leq kn$, where k is the cardinality of the optimal partition. ∎

The next corollary shows that, in any minimal overlap strong optimal partition, all crossing sets of diameter one occur 'close' to the end of the graph.

Corollary 5.1 (Biconvex Matching) *Let $\mathcal{Z}$ be a minimal overlap strong optimal partition that contains two crossing sets, Z_a and Z_b, of diameter one each. If $Z_a \bigcup Z_b$ induces an R-Estar (see lemma 5.4) then all vertices in the set $\{x, y : x_1 \leq x \leq t^X_{ab}, y_1 \leq y \leq t^Y_{ab}\}$ are either in a cardinality one set or in a cardinality two set that crosses one or both of Z_a and Z_b. Otherwise $Z_a \bigcup Z_b$ induces an L-Estar and all vertices in the set $\{x, y : s^X_{ab} \leq x \leq x_p, s^Y_{ab} \leq y \leq y_q\}$ are either in a cardinality one set or in a cardinality two set that crosses one or both of Z_a and Z_b.*

Proof : Let $\mathcal{Z}$ be a minimal overlap strong optimal partition. Let Z_a and Z_b be two crossing sets in $\mathcal{Z}$ that have diameter one each. Note that no set overlaps any of Z_a or Z_b (since $\mathcal{Z}$ is a minimal overlap partition). Assume that $Z_a \bigcup Z_b$ is an R-Estar (see lemma 5.4). Any set that crosses one or both of Z_a or Z_b has diameter one (lemma 5.4). Therefore every vertex in $R(Z_a \bigcup Z_b)$ is either in a cardinality one set or in a cardinality two set that crosses one or both of Z_a and Z_b.

Let x_i be a vertex smaller than $R^X(Z_a \bigcup Z_b)$. x_i has no neighbors smaller than $R(Z_a \bigcup Z_b)$, otherwise $Z_a \bigcup Z_b$ does not induce an R-Estar. Therefore x_i is in a cardinality one set or a cardinality two set that crosses one or both of Z_a or Z_b. This follows for all vertices smaller than $R^Y(Z_a \bigcup Z_b)$ as well.

If $Z_a \bigcup Z_b$ induces an L-Estar, then a similar argument shows that every vertex in or larger than $R(Z_a \bigcup Z_b)$ is in a cardinality one or a cardinality two set that crosses one or both of Z_a and Z_b. ∎

We now have some necessary proofs before proceeding to prove properties of the desired minimal overlap optimal partition. In the remainder of this section, we consider connected sets of vertices Z_a, Z_b, Z_c, and Z'_a, that, unless explicitly specified, are not necessarily members of any optimal partition.

Proposition 5.5 *If $diam(Z_a - \{s^X_a\}) > diam(Z_a)$ then either Z_a is a diameter two X-star or an Estar such that s^X_a is in every center of Z_a, or $diam(Z_a) > 3$ and s^Y_a is adjacent to s^X_a and has no other neighbors in Z_a.*

Proof : If Z_a is a star then it must be true that s^X_a is the center and Z_a has diameter two. If Z_a is an Estar such that s^X_a is not in some center then $diam(Z_a - \{s^X_a\}) \leq diam(Z_a)$. Therefore if Z_a is a star or an Estar then it satisfies the claim.

Else Z_a has diameter greater than three. Let $Z'_a = Z_a - \{s^X_a\}$. Let s'^X_a be the smallest index vertex in Z'^X_a.

Suppose s_a^Y has a neighbor in Z_a'. Since Z_a is not an Estar then t_a^Y is not adjacent to s_a^X (otherwise Z_a is a star, or by biconvex S-ordering Z_a is an Estar with center $(prev(s_a^Y), next(t_a^X)))$. Therefore all of s_a^Y,$s_a'^X$, t_a^Y, and t_a^X have neighbors in Z_a'. Therefore by lemma 4.7 Z_a' has a diameter less than or equal to that of Z_a, a contradiction.

Therefore if Z_a is not an Estar then s_a^Y has no neighbors in Z_a other than s_a^X. ■

Note that the above proposition not only applies for s_a^X, but also applies for each of s_a^Y, t_a^X, and t_a^Y, by the symmetry in biconvex graphs.

Proposition 5.6 *If for some set Z_a, there is a set $T \subset Z_a$, such that $|T| > 1$, $diam(Z_a - T) \leq diam(Z_a)$, $s_a^X \in T$, T consists of consecutive vertices in Z_a, $T^Y = \phi$, and $t_a^X \notin T$, then $diam(Z_a - \{s_a^X\}) \leq diam(Z_a)$.*

Proof : Let $s_a'^X$ be the smallest index vertex in $Z_a^X - T$. Suppose $diam(Z_a - \{s_a^X\}) > diam(Z_a)$ then by proposition 5.5 and since Z_a cannot be a diameter two X-star, therefore one of the following is true.

Z_a is an Estar with s_a^X in every center, then $diam(Z_a - T) > diam(Z_a)$, a contradiction to our choice of T.

Otherwise s_a^Y is adjacent to s_a^X and has no other neighbors, and again, $diam(Z_a - T) > diam(Z_a)$, a contradiction to our choice of T. ■

Proposition 5.7 *Let $\mathcal{Z}$ be a minimal overlap strong optimal partition, then for any set, Z_a, in $\mathcal{Z}$, the following is true.*
(a) If each of s_a^X and t_a^X (or each of s_a^Y and t_a^Y) is in the range of some set other than Z_a, then Z_a is a diameter two X-star (resp. Y-star) or Z_a has diameter greater than three.
(b) If each of s_a^X and s_a^Y (or each of t_a^X and t_a^Y) is in the range of some set other than Z_a, then Z_a is an L-Estar (resp. R-Estar).
(c) No more than two of $s_a^X, s_a^Y, t_a^X, t_a^Y$ are each in the range of any set other than Z_a.

Proof: If Z_a has diameter smaller than two, then no vertex in Z_a can be in the range of any set, otherwise $\mathcal{Z}$ is not a minimal overlap partition. Therefore $diam(Z_a) > 1$.

Since $\mathcal{Z}$ is a minimal overlap optimal partition, therefore if $s_a^X \in R_i$ and $t_a^X \in R_j$, for some $i, j \neq a$, then $diam(Z_a - \{s_a^X\}) > diam(Z_a)$ and $diam(Z_a - \{t_a^X\}) > diam(Z_a)$ (by proposition 5.6). Part (a) now follows directly from proposition 5.5 since it cannot be true that $diam(Z_a) \leq 3$, and both s_a^X and t_a^X are distinct, or both s_a^Y and t_a^Y are distinct, and are in every center of Z_a.

To see that (b) is true suppose to the contrary that it is not. Since $\mathcal{Z}$ is a minimal overlap optimal partition, therefore if $s_a^X \in R_i$ and $s_a^Y \in R_j$, for some $i, j \neq a$, then $diam(Z_a - \{s_a^X\}) > diam(Z_a)$ and $diam(Z_a - \{s_a^Y\}) > diam(Z_a)$ (by proposition 5.6). Z_a cannot be a diameter two X-star or Y-star otherwise s_a^Y or s_a^X, respectively, may be removed from Z_a

and $\mathcal{Z}$ is not a minimal overlap partition. Therefore Z_a has diameter greater than three, and by proposition 5.5 both s_a^Y and s_a^X are adjacent and have no other neighbors in Z_a. But since $diam(Z_a) > 1$ this implies that Z_a is not connected, a contradiction to the minimality of $\mathcal{Z}$.

Part (c) now follows from (a) and (b). ∎

5.3 Structure of Optimal Partition

We are now ready to show how some overlap patterns are eliminated, or may be eliminated, in a minimal overlap strong optimal partition. We already proved that in any minimal overlap strong optimal partition crossing sets are limited to sets of equal diameter smaller than three (lemma 5.2), and occur at the 'end' of the graph (corollary 5.1 and implied by lemma 5.3). In the next few lemmas we show that in any minimal overlap strong optimal partition set pairs that form pattern 1 always contain a diameter two star (lemma 5.5), those that form pattern 2 have diameters greater than two and always contain an Estar of special properties (lemma 5.6), those that form pattern 3 always consist of two stars (lemma 5.7), and sets that form pattern 4 always have diameter three (lemma 5.8). In addition there exists a minimal overlap strong optimal partition in which overlap patterns 2 and 4 disappear (lemma 5.9 and lemma 5.10 respectively), and all overlaps and crossings occur at the 'ends' of the graph and always include diameter two stars (lemma 5.11, lemma 5.13, and lemma 5.12). Theorem 5.3 summarizes all these cases.

Let $\mathcal{Z}$ be a minimal overlap strong optimal partition for the biconvex graph $G = (X, Y, E)$ that has a biconvex S-ordering. We will study each pattern in turn.

We start by examining overlap pattern 1.

Lemma 5.5 (Pattern 1: First lemma) *Let $\mathcal{Z}$ be a minimal overlap optimal partition. For any pair $Z_a, Z_b \in \mathcal{Z}$ such that $Z_b^Y \subset R_a^Y$ and $Z_a^X \not\subset R_b^X$ (or $Z_b^X \subset R_a^X$ and $Z_a^Y \not\subset R_b^Y$) (pattern 1), the following is true.*
(a) $Z_b \bigcap R_a = Z_b^Y$ (respectively $Z_b \bigcap R_a = Z_b^X$).
(b) Z_b is a diameter two Y-star (resp. X-star).
(c) If R_b^X (resp. R_b^Y) intersects the range of any set Z_i then R_i^X is a subset of R_b^X (resp. R_i^Y is a subset of R_b^Y).

Proof: $Z_b^X \not\subset R_a^X$ since otherwise the partition is not optimal (see proposition 5.4). Since $Z_a^X \not\subset R_b^X$ then we may assume that $s_b^X < s_a^X$.

We prove parts (a) and (b) first.

Suppose $s_b^Y \neq t_b^Y$ then Z_b is not an Estar (by proposition 5.7).

Assume that Z_b is not a diameter two X-star, then by proposition 5.5 Z_b has a diameter greater than three. Therefore $t_b^X \neq s_b^X$. By proposition 5.5 t_b^X has no neighbor in Z_b other than t_b^Y. But by proposition 5.3 and convexity s_b^Y is adjacent to all vertices in Z_b^X, and in particular to t_b^X, a contradiction. Therefore $s_b^Y = t_b^Y$. By the definition of pattern 1, Z_b cannot be an Estar, therefore Z_b is a diameter two star (see figure 5.5).

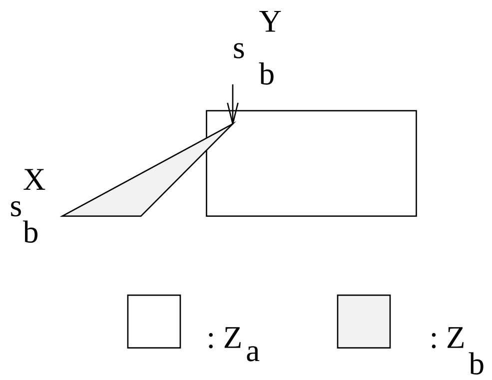

Figure 5.5: (Z_a, Z_b) **minimally overlap in pattern 1**

Part (c) follows directly from the definition of minimally overlapping set pairs and the fact that $diam(Z_b - \{x\}) \leq diam(Z_b)$ for any $x \in Z_b^X$. ■

Definition 5.4: Let (Z_a, Z_b) minimally overlap in pattern 1 such that Z_b is a diameter two star and $Z_b \bigcap R_a$ is the center of Z_b, then Z_b is called the *trailer* with respect to Z_a.

Note that one only of Z_a or Z_b is a trailer with respect to the other.

The next corollary proves that no two trailers of the same type (Y-star or X-star) occur at the same end of any set.

Corollary 5.2 (Two Trailers) *Let $\mathcal{Z}$ be a minimal overlap strong optimal partition that contains a pair, (Z_a, Z_b), that overlaps in pattern 1 such that Z_b is the trailer. Suppose $\mathcal{Z}$ contains another pair (Z_a, Z_c) that overlaps in pattern 1 such that Z_c is the trailer with respect to Z_a. The following is true: if both Z_c and Z_b are Y-stars then $Z_c^X < Z_a^X < Z_b^X$ or $Z_b^X < Z_a^X < Z_c^X$, and if both Z_c and Z_b are X-stars then $Z_c^Y < Z_a^Y < Z_b^Y$ or $Z_b^Y < Z_a^Y < Z_c^Y$.*

Proof: Let the pairs (Z_a, Z_b) and (Z_a, Z_c) minimally overlap in pattern 1 such that both Z_b and Z_c are trailers with respect to Z_a. Suppose both Z_b and Z_c are Y-stars (the other case follows similarly). We may assume, as well, that $Z_b^X < Z_a^X$.

Assume to the contrary that $Z_c^X < Z_a^X$. Then both centers of Z_b and Z_c are in R_a, by proposition 5.3 both have neighbors in R_a, and by convexity one of the two centers is adjacent to all vertices in $(Z_c \bigcup Z_b)^X$. It follows, by proposition 5.4, that there is a smaller partition, a contradiction. Therefore it must be true that $Z_b^X < Z_a^X < Z_c^X$. ■

Lemma 5.6 (Pattern 2: First lemma) *Let $\mathcal{Z}$ be a minimal overlap strong optimal partition that contains a pair of sets (Z_a, Z_b) that form pattern 2. Then the following is true.*

(a) Both sets have diameter greater than two.

(b) At most one of $(R_a \bigcap R_b)^Y$ *and* $(R_a \bigcap R_b)^X$ *is non-empty.*

(c) Furthermore if $(R_a \bigcap R_b)^X \neq \phi$ *then* Z_b *is an Estar with* s_b^X *in every center or* Z_a *is an Estar with* t_a^X *in every center, else* $(R_a \bigcap R_b)^Y \neq \phi$*, and* Z_b *is an Estar with* s_b^Y *in every center or* Z_a *is an Estar with* t_a^Y *in every center.*

Proof: Suppose $(Z_a - Z_b) < Z_b$ and $(R_a \bigcap R_b)^X \neq \phi$ (the other cases follow similarly).

Since (Z_a, Z_b) minimally overlap then $diam(Z_b - \{s_b^X\}) > diam(Z_b)$, and $diam(Z_a - \{t_a^X\}) > diam(Z_a)$. Since none of Z_a or Z_b is a star (by proposition 5.5 and the definition of pattern 2), therefore by proposition 5.5, Z_a is an Estar such that t_a^X is in every center of Z_a or t_a^Y is adjacent to t_a^X and has no other neighbors in Z_a, and Z_b is an Estar such that s_b^X is in every center of Z_b or s_b^Y is adjacent to s_b^X and has no other neighbors in Z_b. Therefore both diameters have to be greater than two.

We now prove part (b). If, in addition, $(R_a \bigcap R_b)^Y \neq \phi$ then $diam(Z_b - \{s_b^Y\}) > diam(Z_b)$ and $diam(Z_a - \{t_b^Y\}) > diam(Z_a)$. By proposition 5.7 Z_a is an R-Estar and Z_b is an L-Estar. It follows by lemma 4.5 that we have a smaller optimal partition, a contradiction to the minimality of $\mathcal{Z}$.

Therefore at most one of $(R_a \bigcap R_b)^X$ and $(R_a \bigcap R_b)^Y$ is non-empty.

It remains to prove part (c). From part (b) we have $(R_a \bigcap R_b)^Y = \phi$. Suppose neither Z_a nor Z_b is an Estar (see figure 5.6).

Then there is an edge, $(t_b^Y, prev(t_b^Y))$ or $(t_b^X, prev(t_b^X))$, with both endpoints larger than or equal to t_a^X or $next(s_b^X)$, otherwise Z_b is an Estar with center $(prev(t_b^Y), prev(t_b^X))$ (by S-ordering, proposition 5.3 and biconvexity since $prev(t_b^Y)$ would be in R_a and $prev(t_b^X)$ would be adjacent to s_b^X). Similarly, there is an edge, $(s_a^Y, prev(s_a^Y))$ or $(s_a^X, prev(s_a^X))$, with both endpoints smaller than or equal to s_b^X or $prev(t_a^X)$.

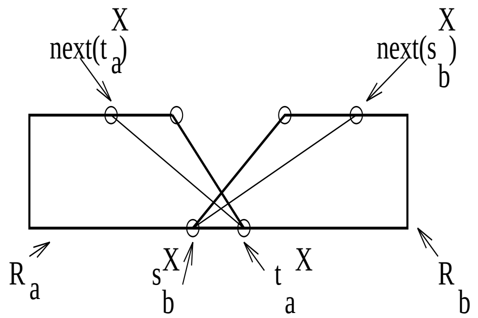

Figure 5.6: (Z_a, Z_b) **minimally overlap in pattern 2**

Therefore s_b^X is adjacent to all neighbors of t_a^X in Z_a, and t_a^X is adjacent to all neighbors of s_b^X in Z_b, by lemma 4.2 (see figure 5.6). Consequently $diam(Z_a - \{t_a^X\} \bigcup \{s_b^X\}) \leq diam(Z_a)$ and $diam(Z_b - \{s_b^X\} \bigcup \{t_a^X\}) \leq diam(Z_b)$, which contradicts the fact that (Z_a, Z_b) minimally overlap.

Therefore at least one of Z_a or Z_b is an Estar. If Z_a is an Estar, then by proposition 5.5 t_a^X is in every center of Z_a. Similarly if Z_b is an Estar then s_b^X is in every center of Z_b. ■

Notation: Let Z_a and Z_b be any two sets in $\mathcal{Z}$.

Let s^X_{ab} and s^Y_{ab} be the smallest index vertices in $Z_a \bigcup Z_b$, and t^X_{ab} and t^Y_{ab} be the largest index vertices in $Z_a \bigcup Z_b$.

Let $R_{ab} = R_a \bigcup R_b$.

Lemma 5.7 (Pattern 3: First lemma) *Let $\mathcal{Z}$ be a minimal overlap strong optimal partition. Any pair (Z_a, Z_b) in $\mathcal{Z}$ that forms pattern 3 satisfies the following.*
(a) Each of Z_a and Z_b is a diameter two star, and $Z_a \bigcup Z_b$ induces an Estar.
(b) At least one of (s^X_{ab}, s^Y_{ab}) or (t^X_{ab}, t^Y_{ab}) is not in E.

Proof : From lemma 5.1 we know that $diam(Z_a) = diam(Z_b)$. Let $D = diam(Z_a) = diam(Z_b)$. $D > 1$ by the definition of the pattern. Furthermore by lemma 5.1 $D \neq 3$.

We consider two cases: $D = 2$ and $D > 3$. In both cases we show that if $\mathcal{Z}$ does not satisfy the lemma, then it does not satisfy theorem 5.1, and is thus not strong, a contradiction.

Suppose $R^X_a \subset R^X_b$ and $R^Y_b \subset R^Y_a$.

Case 1: $D = 2$

If both sets are diameter two sets then $< Z_a \bigcup Z_b >$ is an Estar by lemma 5.1. By proposition 5.7 and the fact that both diameters are smaller than three, it must be true that $s^X_a = t^X_a$ and $s^Y_b = t^Y_b$. Therefore if Z_a and Z_b form pattern 3 then each is a diameter two star. Furthermore by the definition of pattern 3, Z_a is an X-star while Z_b is a Y-star. Therefore part (a) is true for this case.

We proceed to prove part (b). Now suppose both (s^X_b, s^Y_a) and (t^X_b, t^Y_a) are in E (figure 5.7 a). Let $Z'_a = \{x, y : x, y \in Z_a \bigcup Z_b, x \leq s^X_a, y \leq s^Y_b\}$, and $Z'_b = (Z_a \bigcup Z_b) - Z'_a$. Both sets have diameter smaller than or equal to two. But since both s^X_a and s^Y_b are in $R_a \bigcap R_b$, then by theorem 5.1 this implies that $\mathcal{Z}$ is not strong, a contradiction.

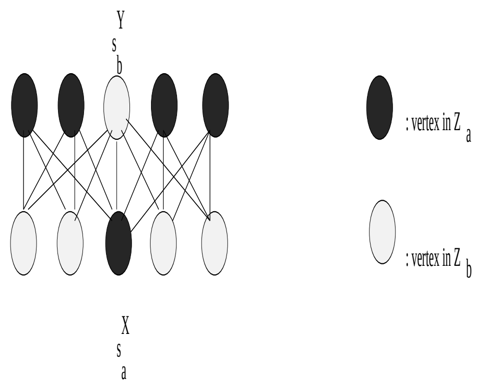

Figure 5.7a: (Z_a, Z_b) **forms pattern 3,** $D = 2$, **and both endpoints of** $Z_a \bigcup Z_b$ **are adjacent**

Case 2: $D > 3$

Since (Z_a, Z_b) minimally overlap and $D > 3$, then by proposition 5.5 t^Y_a is adjacent to t^X_a only in Z_a, and t^X_b is adjacent to t^Y_b only in Z_b. Similarly, s^X_b is adjacent to s^Y_b only in Z_b, and s_a^Y is adjacent to s_a^X only in Z_a.

Since (s_a^Y, s_a^X) crosses (s_b^Y, s_b^X), the edges (t_b^X, t_b^Y) and (t_a^X, t_a^Y) have both endpoints larger than s_a^X or s_b^Y, therefore by lemma 4.2 the edge (s_b^Y, s_a^X) exists. By the same reasoning the edge (t_a^X, t_b^Y) exists. All four vertices s_a^X, s_b^Y, t_a^X, and t_b^Y are in $R_a \bigcap R_b$ (see figure 5.7 b).
Let $S = \{x, y : x, y \in Z_a \bigcup Z_b, t_a^X \leq x \leq t_b^X, t_b^Y \leq y \leq t_a^Y\}$. Since the edge (t_a^X, t_b^Y) exists, then S is an L-Estar or has diameter two.

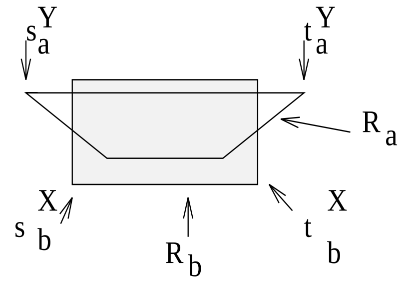

Figure 5.7b: Z_a, Z_b overlap in pattern 3, $D > 3$

Let $Z_a^1 = S \bigcup \{v : v \in Z_a \bigcup Z_b, prev(v/Z_a \bigcup Z_b) \in S\}$, i.e. Z_a^1 consists of those vertices of $Z_a \bigcup Z_b$ that are either in S or have no neighbors, in $Z_a \bigcup Z_b$, outside of S. Let $Z_a^{'} = (Z_a \bigcup Z_b) \bigcap R(Z_a^1)$, i.e. $Z_a^{'}$ consists of those vertices of $Z_a \bigcup Z_b$ that are in the range of Z_a^1. Z_a^1 has diameter smaller than or equal to three by lemma 4.5, if S is an L-Estar, or by corollary 4.6 if S has diameter two. It follows, by corollary 4.5, that $Z_a^{'}$ has diameter less than or equal to three.

Let $s_a^{'Y}$ and $s_a^{'X}$ be the smallest index vertices in $Z_a^{'}$. $s_a^{'Y} > s_b^Y$ and $s_a^{'X} > s_a^X$ (since both s_b^Y and s_a^X are not in S and have neighbors outside of S).

Let $Z_b^{'} = (Z_a \bigcup Z_b) - Z_a^{'}$. Let $t_b^{'Y}$ and $t_b^{'X}$ be the largest index vertices in $Z_b^{'}$.

Therefore both $t_b^{'Y}$ and $t_b^{'X}$ are in $R_a \bigcap R_b$, and both $prev(t_b^{'X})$ and $prev(t_b^{'Y})$ are in $Z_b^{'}$. In fact $t_b^{'Y}$ is adjacent to $t_b^{'X}$ (by lemma 4.2 since there is an edge with both endpoints greater than or equal to $t_b^{'X}$ or $t_b^{'Y}$, e.g. the edge (t_a^X, t_b^Y)). We now show that $Z_b^{'}$ has diameter not larger than D. Let $Z_{b1} = Z_b^{'} \bigcap (Z_b \bigcup (Z_a \bigcap R_b))$, and $Z_{a1} = Z_b^{'} \bigcap (Z_a \bigcup (Z_b \bigcap R_a))$, i.e. both Z_{a1} and Z_{b1} consist entirely of vertices from $Z_b^{'}$, both contain vertices of $Z_b^{'}$ that are in $R_a \bigcap R_b$, in addition Z_{a1} contains the remaining vertices of Z_a and Z_{b1} contains the remaining vertices of Z_b. Therefore $Z_b^{'} = Z_{a1} \bigcup Z_{b1}$. Both $t_b^{'Y}$ and $t_b^{'X}$ are in each of Z_{a1} and Z_{b1} and have neighbors in each set. s_a^Y and s_a^X are the smallest index vertices in Z_{a1}, while s_b^X and s_b^Y are the smallest index vertices in Z_{b1}. By lemma 4.7 and corollary 4.5, $diam(Z_{b1}) \leq D$ and $diam(Z_{a1}) \leq D$. Therefore $dist(t_b^{'Y}, s_a^Y), dist(t_b^{'Y}, s_b^X), dist(t_b^{'X}, s_a^Y), dist(t_b^{'X}, s_b^X)$, where all distances are with respect to $Z_b^{'}$, are all not larger than D. Since also, by S-ordering of the vertices, $dist(s_a^Y, s_b^X) \leq 3 < D$, it follows (by corollary 4.4) that $diam(Z_b^{'}) \leq D$.

Since both $t_b^{'X}$ and $t_b^{'Y}$ are in $R_a \bigcap R_b$, $diam(Z_a^{'}), diam(Z_b^{'}) \leq D$, and $Z_a^{'}$ does not overlap $Z_b^{'}$, then by theorem 5.1, $\mathcal{Z}$ is not strong, a contradiction. ∎

Lemma 5.8 (Pattern 4: First Lemma) *Let $\mathcal{Z}$ be a minimal overlap strong optimal partition that contains a pair of sets (Z_a, Z_b) that form pattern 4, then the following is true.*
(a) Exactly one of $(R_a \bigcap R_b)^Y$ and $(R_a \bigcap R_b)^X$ is non-empty.
(b) Both sets are Estars, but none is an L-Estar or an R-Estar.

Proof: Let (Z_a, Z_b) form pattern 4. Suppose $(R_a \bigcap R_b)^X \neq \phi$ and $s_a^X < s_b^X$. From lemma 5.1 we know that $diam(Z_a) = diam(Z_b)$. Let $D = diam(Z_a) = diam(Z_b)$.

First we show that $D \neq 2$. If $D = 2$ then $diam(Z_a \bigcup \{s_b^X\}) = 2$, $diam(Z_b - \{s_b^X\}) \leq 2$, and hence $\mathcal{Z}$ is not a minimal overlap partition. Thus $D > 2$.

Suppose now that both $(R_a \bigcap R_b)^Y$ and $(R_a \bigcap R_b)^X$ are non-empty.

Either Z_b is an Estar and s_b^X is in every center of Z_b or s_b^Y is adjacent to s_b^X only in Z_b (by proposition 5.5 and the fact that Z_a and Z_b minimally overlap). Therefore in either case the edge (s_b^Y, s_b^X) exists. Similarly the edges (t_a^Y, t_a^X), (s_a^X, s_a^Y) and (t_b^Y, t_b^X) exist (see figure 5.8 a). Since (s_a^X, s_a^Y) crosses (s_b^Y, s_b^X), and the edge (t_a^Y, t_a^X) has both endpoints larger than s_b^X or s_a^Y, therefore by lemma 4.2 the edge (s_a^Y, s_b^X) exists.

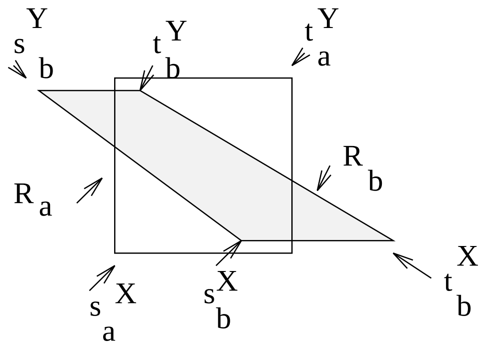

Figure 5.8a: Z_a, Z_b **overlap in pattern 4**

Let $S = \{x, y : x, y \in Z_a \bigcup Z_b, x \leq s_b^X, y \leq s_a^Y\}$. Therefore S is either an R-Estar or has diameter two. Let $Z_a^1 = S \bigcup \{v : v \in Z_a \bigcup Z_b, next(v/Z_a \bigcup Z_b) \in S\}$; i.e. Z_a^1 consists of those vertices of $Z_a \bigcup Z_b$ that are either in S or have no neighbors outside of S. Let $Z_a^{'} = (Z_a \bigcup Z_b) \bigcap R(Z_a^1)$; i.e. $Z_a^{'}$ consists of those vertices of $Z_a \bigcup Z_b$ that are in the range of Z_a^1. Therefore $diam(Z_a^{'}) \leq 3$ (by lemma 4.5 and corollary 4.5).

Let $Z_b^{'} = (Z_a \bigcup Z_b) - Z_a^{'}$. Let $s_b^{'Y}$ and $s_b^{'X}$ be the first vertices in $Z_b^{'}$.

Since both t_a^X and t_b^Y are not in S and have neighbors outside of S, then $s_b^{'Y}$ and $s_b^{'X}$ are in $R_a \bigcap R_b$ and have neighbors in $Z_b^{'}$ (by our choice of Z_a^1). By lemma 4.2 and since the edge (s_a^Y, s_b^X) has both endpoints smaller than $s_b^{'Y}$ or $s_b^{'X}$, therefore $(s_b^{'Y}, s_b^{'X})$ is in E.

We now show that $Z_b^{'}$ has diameter not larger than D. Let $Z_{b1} = Z_b^{'} \bigcap (Z_b \bigcup (Z_a \bigcap R_b))$, and $Z_{a1} = Z_b^{'} \bigcap (Z_a \bigcup (Z_b \bigcap R_a))$, i.e. both Z_{a1} and Z_{b1} consist entirely of vertices from $Z_b^{'}$, both contain vertices of $Z_b^{'}$ that are in $R_a \bigcap R_b$, in addition Z_{a1} contains the remaining vertices of Z_a and Z_{b1} contains the remaining vertices of Z_b. Therefore $Z_b^{'} = Z_{a1} \bigcup Z_{b1}$. Both $s_b^{'Y}$ and $s_b^{'X}$ are in each of Z_{a1} and Z_{b1} and have neighbors in each set. t_a^Y and t_a^X are the largest index vertices in Z_{a1}, while t_b^X and t_b^Y are the largest index vertices in Z_{b1}. By lemma 4.7 and corollary 4.5, $diam(Z_{b1}) \leq D$ and $diam(Z_{a1}) \leq D$. Therefore

$dist(s_b'^Y, t_a^Y), dist(s_b'^Y, t_b^X), dist(s_b'^X, t_a^Y), dist(s_b'^X, t_b^X)$, where all distances are with respect to Z_b', are all not larger than D. Since also, by S-order of the vertices, $dist(t_a^Y, t_b^X) \leq 3 < D$. It follows (by corollary 4.4) that $diam(Z_b') \leq D$.

Since both $s_b'^X$ and $s_b'^Y$ are in $R_a \bigcap R_b$, then by theorem 5.1, $\mathcal{Z}$ is not strong, a contradiction. This completes the proof for part (a).

Now we prove part (b). Suppose $(R_a \bigcap R_b)^Y = \phi$ (see figure 5.8 b). Since (Z_a, Z_b) minimally overlap then, by proposition 5.5, Z_b is an Estar and s_b^X is in every center or s_b^Y is adjacent to s_b^X and has no other neighbors in Z_b.

Therefore in either case the edge (s_b^Y, s_b^X) exists. $s_b^X \in R_a$, and therefore is adjacent to all vertices in Z_b^Y (by proposition 5.3 and convexity). Therefore Z_b is an Estar with center $(s_b^X, next(t_b^X/Z_b))$ (or with center (s_b^X, y) where y is any neighbor of t_b^X in Z_b, by convexity y is adjacent to all vertices in Z_b^X). Similarly Z_a is an Estar with center $(t_a^X, prev(s_a^X/Z_a))$.

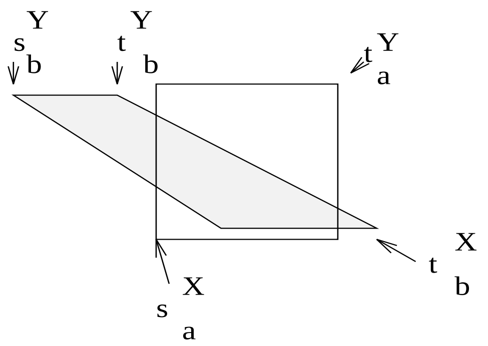

Figure 5.8b: Z_a, Z_b overlap in pattern 4, $Z_b^Y < Z_a^Y$

Suppose that Z_a is an R-Estar, then s_b^X is adjacent to t_a^Y, and by convexity to all vertices in $(Z_a \bigcup Z_b)^Y$, and the edges (s_a^X, t_a^Y) and $(next(t_b^X/Z_b), t_b^X)$ cross. Therefore, by biconvex S-ordering, $Z_a \bigcup Z_b$ is an Estar with center in the set $\{s_b^X, t_a^Y, next(t_b^X/Z_b)\}$. If Z_a is an L-Estar then the edge (s_a^X, t_a^Y) crosses the edges $(next(t_b^X/Z_b), t_b^X)$ and (s_b^Y, s_b^X). Therefore, by biconvex S-ordering, $Z_a \bigcup Z_b$ is an Estar with center in the set $\{s_a^X, s_b^X, t_a^Y, next(t_b^X/Z_b)\}$. Therefore Z_a is neither an L-Estar nor an R-Estar. A similar argument proves that Z_b can neither be an L-Estar nor an R-Estar.

This completes the proof for part (b) and the lemma. ■

In the next few lemmas we show that there exists a minimal overlap strong optimal partition in which no set pair forms pattern 2 (lemma 5.9) or pattern 4 (lemma 5.10), all set pairs that form pattern 3 occur 'close' to the end of the graph (lemma 5.12), and all trailers contain one of the end vertices of the graph (lemma 5.13). Note that any minimal overlap strong optimal partition satisfies lemmas 5.5 through 5.8, and corollary 5.1. We will start with a minimal overlap strong optimal partition containing the minimum number of set pairs that violate conditions of the current lemma. We will choose a pair of sets (Z_a, Z_b) that violate conditions of the lemma, and find a subpartition of $Z_a \bigcup Z_b$ into sets $\{Z_a', Z_b'\}$, such that (Z_a', Z_b') satisfy the conditions of the lemma, $diam(Z_a') \leq diam(Z_a)$, and

$diam(Z_b^{'}) \leq diam(Z_b)$. Thus $\mathcal{Z}^{'} = \mathcal{Z} - \{Z_a, Z_b\} \bigcup \{Z_a^{'}, Z_b^{'}\}$ is an optimal partition. We finally show that $\mathcal{Z}^{'}$ is a minimal overlap strong optimal partition with fewer set pairs that violate that lemma, hence a contradiction to our choice of $\mathcal{Z}$.

In the next few lemmas we will use the next, easy to see, proposition frequently in our arguments.

Proposition 5.8 *Let (Z_i, Z_j) be two sets in an optimal partition that overlap in pattern 1 such that Z_i is a diameter two star, and $Z_i \bigcap R_j$ equals the center of Z_i, then (Z_i, Z_j) minimally overlap.*

Lemma 5.9 (Pattern 2) *There is a minimal overlap strong optimal partition in which no set pair forms pattern 2.*

Proof: Let $\mathcal{Z}$ be a minimal overlap strong optimal partition that has the minimum number of set pairs that violate this lemma. Let Z_a and Z_b in $\mathcal{Z}$, be a pair that minimally overlaps and forms pattern 2. Suppose $(R_b - R_a)^X > R_a^X$, $(R_b - R_a)^Y > R_a^Y$.

Suppose $(R_a \bigcap R_b)^X \neq \phi$, and that Z_a is an Estar with center (t_a^X, y_a^*) (recall lemma 5.6, see figure 5.9 a).

The proof progresses in two steps. Firstly we show that there is a subpartition of $Z_a \bigcup Z_b$, $\{Z_a^{'}, Z_b^{'}\}$, such that $Z_a^{'}$ does not overlap $Z_b^{'}$ or they minimally overlap in pattern 1, $diam(Z_a^{'}) \leq diam(Z_a)$, and $diam(Z_b^{'}) \leq diam(Z_b)$ (step (a) below). Therefore the partition $\mathcal{Z}^{'} = \mathcal{Z} - \{Z_a, Z_b\} \bigcup \{Z_a^{'}, Z_b^{'}\}$ is optimal, since $|\mathcal{Z}^{'}| = |\mathcal{Z}|$ and $f(\mathcal{Z}^{'}) \leq f(\mathcal{Z})$. We then show that the new partition is a minimal overlap strong partition that has fewer set pairs that violate the lemma (step (b)), hence a contradiction to our choice of $\mathcal{Z}$.

(a) We first show that there is a minimal overlap optimal partition, $\{Z_a^{'}, Z_b^{'}\}$, of $Z_a \bigcup Z_b$, such that $Z_a^{'}$ and $Z_b^{'}$ minimally overlap in pattern 1 or do not overlap.

The edges (t_a^X, s_a^Y) and $(s_b^X, next(s_b^X))$ cross, where $next(s_b^X)$ refers to $next(s_b^X/Z_b)$. At most one of the edges $(t_a^X, next(s_b^X))$ and (s_b^X, s_a^Y) exists otherwise the pair (Z_a, Z_b) does not minimally overlap (since we can exchange t_a^X and s_b^X between sets Z_a and Z_b, and still have an optimal partition).

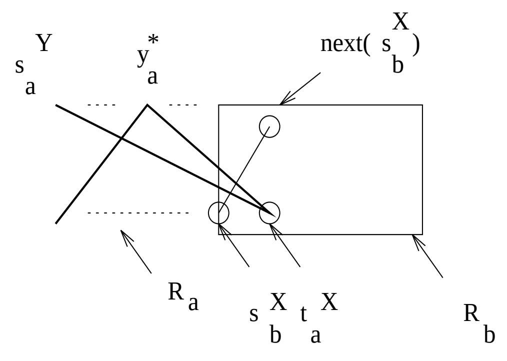

Figure 5.9a: Z_a and Z_b minimally overlap in pattern 2

If $(t_a^X, next(s_b^X))$ is in E then let $x^* = t_a^X$, otherwise let $x^* = s_b^X$. Therefore x^* is adjacent to all y, $s_a^Y \leq y \leq next(s_b^X)$. Note that if Z_b is not an Estar then t_a^X is adjacent to $next(s_b^X)$, and $x^* = t_a^X$ (by lemma 4.2, since there will exist an edge with both endpoints larger than or equal to t_a^X or $next(s_b^X)$).

Let $Z_a^{'} = \{x : x \in (Z_a \bigcup Z_b)^X, x < x^*\}$. If $Z_a^{'}$ has cardinality greater than one then let $Z_a^{'} = Z_a^{'} \bigcup \{y_a^*\}$. Let $Z_b^{'} = (Z_a \bigcup Z_b) - Z_a^{'}$. $Z_a^{'} = \{s_a^X\}$ in which case $Z_a^{'}$ and $Z_b^{'}$ do not overlap, or $diam(Z_a^{'}) = 2$ in which case $Z_a^{'}$ is a Y-star. Therefore $diam(Z_a^{'}) < diam(Z_a)$, and $Z_a^{'}$ and $Z_b^{'}$ minimally overlap in pattern 1 if $y_a^* \neq s_a^Y$ (see figure 5.9 b), otherwise they do not overlap.

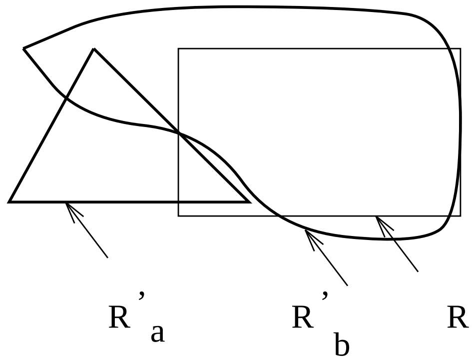

Figure 5.9b: Effect of lemma 5.9

We now show that $diam(Z_b^{'}) = diam(Z_b)$. Note that $R_b^{'X} \subseteq R_b^X$ and $Z_b^Y \subset Z_b^{'Y}$. If Z_b is an Estar with center (s_b^X, y_b^*) then $Z_b^{'}$ is an Estar with center (x^*, y_b^*). Otherwise $diam(Z_b) > 3$, $x^* = t_a^X$, and $dist(y, v) \leq dist(s_b^Y, v)$ for all y in Z_a and v in Z_b. This may be explained as follows. Since s_b^Y has no neighbor in Z_b other than s_b^X (by proposition 5.5), therefore every path from s_b^Y to any vertex v in Z_b passes through s_b^X, if v is in $Z_b^{'}$ then x^* may replace s_b^X in that path. Since every vertex $y \in Z_a$ is adjacent to x^*, therefore $dist(y, v) \leq dist(s_b^Y, v)$, for all v in $Z_b^{'} \bigcap Z_b$. Since, as well, the distance between any two vertices in $Z_b^{'} \bigcap Z_a$, through vertices in $Z_b^{'}$, is smaller than or equal to two, and every other vertex in $Z_b^{'}$ is in R_b and has a neighbor in $Z_b^{'} \bigcap R_b$, therefore $diam(Z_b^{'}) \leq diam(Z_b)$ (see corollary 4.5 and lemma 4.7).

Let $\mathcal{Z}' = \mathcal{Z} - \{Z_a, Z_b\} \bigcup \{Z_a^{'}, Z_b^{'}\}$. $\mathcal{Z}'$ is optimal.

(b) It remains to show that $\mathcal{Z}'$ is a minimal overlap strong partition that has fewer set pairs that violate this lemma. Any set that overlaps or crosses one of $Z_a^{'}$ and $Z_b^{'}$, overlaps or crosses one of Z_a and Z_b.

We will consider all possible ways a set, Z_c, can overlap any of Z_a or Z_b, and show in every case that Z_c minimally overlaps each of $Z_a^{'}$ and $Z_b^{'}$ (hence $\mathcal{Z}'$ is a minimal overlap partition), if Z_c overlaps both $Z_a^{'}$ and $Z_b^{'}$ then it overlaps both Z_a and Z_b (hence $\mathcal{Z}'$ is strong), and for every pair $(Z_c, Z_a^{'})$ or $(Z_c, Z_b^{'})$ that violates this lemma there is a pair (Z_c, Z_a) or (Z_c, Z_b) that violates the lemma as well where $c \notin \{a, b\}$. The result is that we have a minimal overlap strong optimal partition with fewer set pairs that violate this lemma, a contradiction to our choice of $\mathcal{Z}$.

No set can cross or form pattern 3 with any of Z_a or Z_b (by lemma 5.4 and lemma 5.7, since both diameters are greater than two by lemma 5.6).

If some set, Z_c, forms pattern 1 with Z_a or Z_b, then Z_c is the trailer (by lemma 5.5 since both Z_a and Z_b are not stars by lemma 5.6), and $Z_c \bigcap (R_a \bigcup R_b) = 1$ by part (c) of lemma 5.5 since it cannot be true that $R_b^Y \subset R_c^Y$, $R_b^X \subset R_c^X$, $R_a^Y \subset R_c^Y$, or $R_a^X \subset R_c^X$ (otherwise (Z_c, Z_a) or (Z_c, Z_b) forms pattern 3 which contradicts lemma 5.7, or Z_a or Z_b is a trailer with respect to Z_c which contradicts lemma 5.9). Note that Z_c cannot cross any of $Z_a^{'}$ or $Z_b^{'}$, otherwise by lemma 5.2 there is a smaller partition. Therefore Z_c overlaps at most one of $Z_a^{'}$ and $Z_b^{'}$ in pattern 1 in accordance with lemma 5.5. Furthermore each of the pairs $(Z_c, Z_a^{'})$ and $(Z_c, Z_b^{'})$ is either strong or minimally forms pattern 1 by proposition 5.8.

Therefore it remains to consider the cases where Z_c forms pattern 2 or pattern 4 with Z_a or Z_b. For that purpose we have the next two claims.

Claim 5.9.1 *Let $\mathcal{Z}$ be a minimal overlap strong optimal partition. $\mathcal{Z}$ cannot contain two pairs of sets (Z_i, Z_j) and (Z_i, Z_l) such that (Z_i, Z_j) minimally overlap in pattern 2 and (Z_i, Z_l) minimally overlap in pattern 4.*

Proof of claim: (Z_i, Z_l) minimally overlap in pattern 4 implies that both Z_i and Z_l are Estars by lemma 5.8. (Z_i, Z_j) minimally overlap in pattern 2, then we may suppose that $(R_i \bigcap R_j)^X \neq \phi$, and $(R_i - R_j) < R_j$ (see lemma 5.6). Then by lemma 5.6 Z_i is an Estar with center (t_i^X, y_i^*) and $Z_i^Y < Z_j^Y$.

Since $t_i^X \in R_j$ then neither s_i^X nor t_i^Y is in R_l otherwise Z_i is not an Estar or is an R-Estar respectively (by proposition 5.7), both contradict lemma 5.8.

Note that Z_l cannot cross or form pattern 1 or pattern 3 with Z_j. This follows by lemma 5.4, lemma 5.5, and lemma 5.8 since they both have diameter greater than two; Z_l is an Estar and by lemma 5.6 Z_j has diameter greater than two.

If $t_i^X \in R_l$ then $s_l^X \in R_i$, $t_l^X > t_i^X$, and $Z_l^Y < Z_i^Y$ (by the definition of pattern 4 and lemma 5.8). Therefore Z_l overlaps Z_j, and either $t_l^X > t_j^X$ and $s_j^X < s_l^X < t_i^X$ (i.e. (Z_l, Z_j) forms pattern 4), or $s_l^X < s_j^X < t_i^X < t_l^X < t_j^X$ (i.e. (Z_l, Z_j) forms pattern 2). Both cases are contradictions; the former case implies that $s_j^X \in R_i$ and $t_j^X \in R_l$ then by proposition 5.7 Z_j cannot be an Estar and that contradicts lemma 5.8, the later case implies a contradiction by the same reasoning since $s_l^X \in R_i$ and $t_l^X \in R_j$.

Suppose now $s_i^Y = y_i^* \in R_l$. Then by the definition of pattern 4 and lemma 5.8 $Z_l^X > Z_i^X$. Since Z_l cannot cross Z_j, therefore $t_i^X < s_l^X < t_j^X < t_l^X$ and (Z_l, Z_j) forms pattern 4. But by proposition 5.7 and since $s_j^X \in R_i$ and $t_j^X \in R_l$, then Z_j cannot be an Estar; that contradicts lemma 5.8.

Therefore Z_l cannot form pattern 4 with Z_i. ■

Therefore, by the above claim, no set forms pattern 4 with Z_a or Z_b, and any set that overlaps Z_a or Z_b does so in pattern 2. We next show that any such set overlaps at most one of Z_a and Z_b.

Claim 5.9.2 *In any minimal overlap strong optimal partition no more than two sets mutually overlap in pattern 2.*

Proof of claim: Suppose not. Let Z_a, Z_b, and Z_c be three sets such that each pair overlaps in pattern 2. Suppose $(Z_b - Z_a) > Z_a$ and $(R_b \bigcap R_a)^X \neq \phi$ (recall lemma 5.6).

If $(Z_c - Z_b) > Z_b$ then $(Z_c - Z_a) > Z_a$ and $(R_c \bigcap R_a)^X \neq \phi$ (otherwise $Z_b^Y \subset R_c^Y$). Therefore $(R_c \bigcap R_b)^Y = \phi$, $Z_b^X \subset (R_c \bigcup R_a)^X$, and s_c^X is adjacent to all vertices in Z_b^Y (by proposition 5.3 since $s_c^X \in R_a$ and has a neighbor greater than t_b^X). Therefore if s_c^Y has no neighbors in Z_c other than s_c^X, then $dist(y,v) \leq dist(s_c^Y, v)$ for all y in Z_b^Y and all v in Z_c, otherwise, by proposition 5.5, Z_c is an Estar with s_c^X in every center and $Z_c \bigcup Z_b^Y$ is an Estar as well. Let $Z_a' = Z_a \bigcup (Z_b \bigcap R_a)$ and $Z_c' = Z_c \bigcup Z_b^Y \bigcup (Z_b \bigcap R_c)$. By corollary 4.6 $diam(Z_a') \leq diam(Z_a)$ and $diam(Z_c') \leq diam(Z_c)$. Therefore $\mathcal{Z} - \{Z_a, Z_b, Z_c\} \bigcup \{Z_a', Z_c'\}$ is a smaller optimal partition, a contradiction.

Similarly if $(Z_c - Z_b) < Z_b$ and $(Z_c - Z_a) > Z_a$ then $(R_c \bigcap R_a)^Y = \phi$ (otherwise both $(R_c \bigcap R_b)^X$ and $(R_c \bigcap R_b)^Y$ are non-empty), Z_c^X is a subset of $(R_a \bigcup R_b)^X$ and s_b^X is adjacent to all vertices in Z_c^Y, and if $(Z_c - Z_b) < Z_b$ and $(Z_c - Z_a) < Z_a$ then $Z_a^X \subset (R_c \bigcup R_b)^X$ and t_c^X is adjacent to all vertices in Z_a^Y. Therefore in both cases there is a smaller optimal partition, a contradiction. ∎

From the above, $\mathcal{Z}'$ is a strong optimal partition. It remains to show that it is a minimal overlap partition with fewer set pairs that violate this lemma. We have one of the following cases.

(i) Suppose Z_c overlaps Z_a in pattern 2, then $(Z_c - Z_a) < Z_a$. Since s_a^X cannot be in R_c (by proposition 5.7 and since Z_a is an Estar), therefore $y_a^* = s_a^Y \in R_c$. But this implies that t_c^Y is adjacent to all vertices in $Z_a'^X$ (by convexity since t_c^Y is adjacent to t_a^X) and that $diam(Z_c \bigcup Z_a') \leq diam(Z_c)$ (by lemma 4.6 and since Z_c and Z_a minimally overlap then t_c^X cannot be removed from Z_c without increasing its diameter), a contradiction to the minimality of $\mathcal{Z}$.

(ii) Suppose Z_c overlaps Z_b in pattern 2, Z_c must be such that $(Z_c - Z_b) > Z_b$ and Z_c does not overlap Z_a. Therefore Z_c forms the same pattern with Z_b' and does not overlap Z_a'. We now show that Z_c minimally overlaps Z_b'.

If $t_b^Y \in R_c$ then Z_b is an Estar whose only center is (s_b^X, t_b^Y), or $diam(Z_b) > 3$ and t_b^X is adjacent to t_b^Y only in Z_b (see proposition 5.7). In any case t_b^X has no neighbor in Z_b other than t_b^Y (otherwise there is a center that does not include t_b^Y). In fact it has no other neighbor in Z_b', otherwise by convexity it is adjacent to all vertices in Z_b. Therefore t_b^Y

cannot be removed from Z_b' without increasing its diameter, and Z_c minimally overlaps Z_b' (see proposition 5.6).

If $t_b^X \in R_c$, then by proposition 5.7 Z_a has diameter greater than three. If Z_c and Z_b' do not minimally overlap, then t_b^Y has a neighbor in $Z_b' \bigcap Z_a$. By convexity t_b^Y is adjacent to t_a^X. Then $Z_a \bigcup Z_b^Y$ is an Estar with center (t_a^X, y_a^*). By corollary 4.6 $diam(Z_a \bigcup Z_b) = 4 \leq diam(Z_a)$, a contradiction.

It follows then that $\mathcal{Z}'$ is a minimal overlap strong optimal partition that has fewer set pairs that violate this lemma, a contradiction to our choice of $\mathcal{Z}$. ∎

We now examine overlap pattern 4 (figure 5.2 d) and show that there is a minimal overlap strong optimal partition that satisfies lemma 5.9, in which no two sets form pattern 4.

Lemma 5.10 (Pattern 4: Second lemma) *There is a minimal overlap strong optimal partition that satisfies the previous lemma in which no set pair forms pattern 4.*

Proof : Let (Z_a, Z_b) minimally overlap in pattern 4. Suppose that $R_a^X \bigcap R_b^X \neq \phi$, and $s_a^X < s_b^X$.

From lemma 5.1 we know that $diam(Z_a) = diam(Z_b)$. Let $D = diam(Z_a) = diam(Z_b)$. By lemma 5.8 $D = 3$, and exactly one of $(R_a \bigcap R_b)^X$ and $(R_a \bigcap R_b)^Y$ is non-empty. Suppose $(R_a \bigcap R_b)^Y = \phi$.

We start with a minimal overlap strong optimal partition, $\mathcal{Z}$, that satisfies the previous lemma, and has the minimum number of set pairs that violate this lemma.

Since $s_b^X \in R_a$, therefore it is adjacent to all vertices in Z_b^Y (by proposition 5.3 and convexity). Therefore Z_b is an Estar with center $(s_b^X, next(t_b^X/Z_b))$ (by convexity and since every neighbor of t_b^X in Z_b is adjacent to s_b^X). Similarly Z_a is an Estar with center $(t_a^X, prev(s_a^X/Z_a))$ (see figure 5.10).

s_b^X is adjacent to all vertices in $(Z_a \bigcup Z_b)^Y$ that are smaller than or equal to $prev(s_a^X/Z_a)$, and t_a^X is adjacent to all vertices in $(Z_a \bigcup Z_b)^Y$ that are larger than $prev(s_a^X/Z_a)$. The edges $(t_b^X, next(t_b^X/Z_b))$ and $(s_a^X, prev(s_a^X/Z_a))$ cross, therefore by biconvex S-ordering one of $next(t_b^X/Z_b)$ and $prev(s_a^X/Z_a)$ is adjacent to all vertices in $(Z_a \bigcup Z_b)^X$ (see figure 5.10).

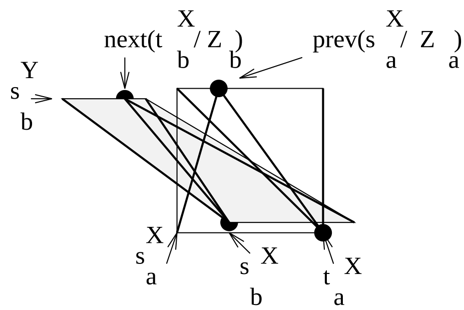

Figure 5.10: Illustration for the proof of lemma 5.10

Let $Z_a' = (Z_a \bigcup Z_b)^X - \{t_a^X\} \bigcup \{y : y \in Z_a \bigcup Z_b, y \leq prev(s_a^X/Z_a)\}$. By the argument in the above paragraph, we know that Z_a' is an Estar with center in the set $\{s_b^X, next(t_b^X/Z_b), prev(s_a^X/Z_a)\}$.

Let $Z_b' = (Z_a \bigcup Z_b) - Z_a'$. By the argument in the above paragraph Z_b' is an X-star. If $|Z_b'| \leq 2$, then let $Z_b' = Z_b'^Y$ and $Z_a' = Z_a' \bigcup Z_b'^X$. Let $\mathcal{Z}' = \mathcal{Z} - \{Z_a, Z_b\} \bigcup \{Z_a', Z_b'\}$; $\mathcal{Z}'$ is optimal.

Therefore Z_a' and Z_b' either do not overlap or they minimally overlap in pattern 1 in accordance with lemma 5.5 (see proposition 5.8).

We now show that $\mathcal{Z}'$ is a minimal overlap strong partition that satisfies the previous lemma, and has fewer set pairs that violate this lemma. Since any set, Z_c, that overlaps or crosses Z_a' or Z_b', crosses or overlaps at least one of Z_a and Z_b, we will consider all cases of overlap or crossing for the pairs (Z_c, Z_a) and (Z_c, Z_b).

By lemma 5.9 no set overlaps Z_a or Z_b in pattern 2.

By lemma 5.4, lemma 5.7, and since both Z_a and Z_b have diameters greater than two, then no set crosses or forms pattern 3 with any of Z_a or Z_b.

If any set, Z_c, forms pattern 1 with Z_a, for instance, then by lemma 5.5, it is a diameter two star such that $Z_c \bigcap R_a$ is the center of the star. Note that Z_c cannot cross any of Z_a' or Z_b', otherwise by lemma 5.2 there is a smaller partition. Since Z_c cannot form pattern 3 with Z_b (by lemma 5.7 since Z_b has diameter greater than two) nor can it form pattern 1 with Z_b such that Z_b is the trailer, then by part (c) of lemma 5.5 it follows that $Z_c \bigcap (R_a \bigcup R_b)$ is the center of Z_c, and Z_c minimally overlaps one only of Z_a' or Z_b' such that Z_c is the trailer (see also proposition 5.8).

Before we proceed to consider the case where Z_c forms pattern 4 with Z_a, we prove the following claim.

Claim 5.10.1 *In any minimal overlap strong optimal partition no more than two sets mutually overlap in pattern 4.*

Proof of claim: Suppose there are sets Z_a, Z_b, and Z_c, such that each pair overlaps in pattern 4. Let s_{abc}^Y, s_{abc}^X, t_{abc}^Y, and t_{abc}^X be the end vertices of $Z_a \bigcup Z_b \bigcup Z_c$. Suppose $s_{abc}^Y \in Z_b$. Then it must be true that $t_{bc}^X \in Z_b$ and $t_{ab}^X \in Z_b$. Consequently $t_{abc}^X \in Z_b$. Similarly both s_{abc}^X and t_{abc}^Y are in the same set.

Suppose s_{abc}^X and t_{abc}^Y are in Z_c.

We may assume, without loss of generality, that $(R_b \bigcap R_a)^Y = \phi$ (see lemma 5.8). Then it must be true that $(R_c \bigcap R_a)^X$ and $(R_c \bigcap R_b)^X$ are both non-empty (otherwise Z_c cannot form pattern 4 with both sets). Therefore $Z_a^X \subset (R_c \bigcup R_b)^X$. Since $s_b^X \in R_c$ and $R_c^Y > R_a^Y$, therefore by proposition 5.3 s_b^X is adjacent to all vertices in Z_a^Y. Now $Z_b \bigcup Z_a^Y$ is an Estar with s_b^X in every center. Let $Z_b = Z_b \bigcup Z_a^Y \bigcup (Z_a^X \bigcap R_b)$ and $Z_c = Z_c \bigcup (Z_a^X \bigcap R_c)$. We thus have a smaller optimal partition, a contradiction. ■

Suppose Z_c forms pattern 4 with Z_a. Z_c cannot cross Z_b (by lemma 5.4), Z_c cannot form pattern 1 with Z_b (since this would imply that $diam(Z_c) = 2$ by lemma 5.5, and $diam(Z_a) = 2$ by lemma 5.1), and Z_c cannot overlap Z_b in pattern 3 or 4 (by lemma 5.2, lemma 5.7 and claim 5.10.1). Therefore Z_c can neither overlap nor cross Z_b. Then it must be true that

$(R_c \bigcap R_a)^Y \neq \phi$, $(R_c \bigcap R_a)^X = \phi$, $s_c^Y > t_b^Y$ and $s_c^X > t_b^X$. Z_a is an Estar with center (t_a^X, s_a^Y), by lemma 5.8. Suppose Z_b has center (s_b^X, y_b^*), and Z_c has center (x_a^*, t_c^Y).

Now consider the set $Z_a \bigcup Z_b \bigcup Z_c$, we will show that it can be partitioned into two Estars, hence a contradiction to the minimality of $\mathcal{Z}$. Let all distances refer to distances in $Z_a \bigcup Z_b \bigcup Z_c$. Note that t_a^X is adjacent to t_c^Y, and by biconvexity t_a^X is adjacent to all vertices between s_a^Y and t_c^Y (inclusive), and t_c^Y is adjacent to all vertices between t_a^X and t_c^X (inclusive). Similarly s_b^X is adjacent to all y, $s_b^Y \leq y \leq s_a^Y$, and s_a^Y is adjacent to all x, $s_a^X \leq x \leq s_b^X$. Let $Z_a' = \{x, y : x, y \in Z_a \bigcup Z_b \bigcup Z_c, x < t_a^X, y \leq s_a^Y\}$ and $Z_b' = \{x, y : x, y \in Z_a \bigcup Z_b \bigcup Z_c, x \geq t_a^X, y > s_a^Y\}$. By the above arguments, Z_a' is an Estar with center (s_b^X, s_a^Y) and Z_b' is an Estar with center (t_a^X, t_c^Y). We now have a contradiction to the minimality of $\mathcal{Z}$.

Therefore $\mathcal{Z}'$ is a minimal overlap strong optimal partition with fewer set pairs that violate this lemma, a contradiction. ∎

Lemma 5.11 (Crossing Partitions: Last Lemma) *There is a minimal overlap strong optimal partition that satisfies all previous lemmas, in which any two diameter two crossing sets, Z_a and Z_b, are stars; one is an X-star, the other is a Y-star, and $< Z_a \bigcup Z_b >$ is an L-Estar or an R-Estar with a center that is composed of the centers of Z_a and Z_b.*

Proof: Suppose Z_a crosses Z_b, both have diameter two, $Z_a^Y < Z_b^Y$, and $Z_a \bigcup Z_b$ induces an L-Estar. Therefore $Z_a \bigcup Z_b$ contains x_p and y_q (by lemma 5.3). By lemma 5.4 no other set crosses Z_a or Z_b.

We first show that if some set, Z_c, overlaps Z_a (or Z_b), then Z_a (resp.Z_b) is a trailer with respect to Z_c.

By lemma 5.9 and lemma 5.10 Z_c cannot overlap Z_a in pattern 2 or pattern 4.

If Z_c overlaps Z_a in pattern 3 then since x_p is in Z_a, therefore Z_a is a Y-star and Z_c is an X-star (see lemma 5.7), but now Z_c either crosses Z_b or overlaps Z_b in pattern 4, both cases are contradictions (to lemma 5.4 and lemma 5.10 respectively).

Therefore any set, Z_c, that overlaps Z_a forms pattern 1 with Z_a. Suppose Z_c is the trailer with respect to Z_a. It cannot be true that Z_c is an X-star and $Z_c^Y > Z_a^Y$, or Z_c is a Y-star and $Z_c^X > Z_b^X$ (otherwise Z_c is a subset of $R(Z_a \bigcup Z_b)$, $Z_a \bigcup Z_b \bigcup Z_c$ induces an L-Estar, and there is a smaller partition by proposition 5.1). Therefore either Z_c is an X-star and $Z_c^Y < Z_a^Y$ but in this case Z_c crosses Z_b, or Z_c is a Y-star and $Z_c^X < Z_b^X$ but in this case by lemma 4.5 and proposition 5.1 there is a smaller partition, a contradiction. Subsequently it must be true that Z_a is a trailer with respect to Z_c. Therefore any set, Z_c, that overlaps Z_a is such that Z_a is a trailer with respect to Z_c.

Now let $Z_a' = \{s_a^Y\} \bigcup (Z_a \bigcup Z_b)^X - \{s_b^X\}$, and $Z_b' = (Z_a \bigcup Z_b) - Z_a'$. Note that both Z_a' and Z_b' must have diameter two. Otherwise there would be a subpartition of $< Z_a \bigcup Z_b >$

into a diameter two and a diameter zero set, and that would imply that $\mathcal{Z}$ is not strong. Therefore Z_a' crosses Z_b'. Z_a' is a diameter two Y-star while Z_b' is a diameter two X-star, and $< Z_a \bigcup Z_b >$ is an L-Estar with center (s_a^Y, s_b^X). $\mathcal{Z}' = \mathcal{Z} - \{Z_a, Z_b\} \bigcup \{Z_a', Z_b'\}$ is optimal.

If some set Z_c overlap Z_a then Z_a is a trailer with respect to Z_c and is a Y-star. Additionally by lemma 5.5 $(R_a \bigcap R_c)^X = \phi$. It follows that Z_a' minimally overlaps Z_c in pattern 1 such that Z_a' is the trailer with respect to Z_c and is a Y-star, and Z_c does not overlap Z_b' unless it overlaps Z_b as well. If Z_c overlaps Z_b then Z_b is a Y-star; a similar argument would show that (Z_c, Z_b') minimally overlap in pattern 1.

Therefore $\mathcal{Z}'$ is a minimal overlap strong partition. ■

We now show that there is a minimal overlap optimal partition in which overlapping sets that form pattern 3 do not overlap other sets and are always towards the 'end' of the graph.

Lemma 5.12 (Pattern 3 and Others) *Let Z_a and Z_b be two sets that form pattern 3. There is a minimal overlap strong optimal partition that satisfies all previous lemmas and the following.*
(a) No other set crosses or overlaps any of Z_a or Z_b.
(b) If (s_{ab}^Y, s_{ab}^X) is not in E then any vertex smaller than R_{ab} is in a cardinality one set and has a neighbor in R_{ab}, and if (t_{ab}^Y, t_{ab}^X) is not in E then any vertex larger than R_{ab} is in a cardinality one set and has a neighbor in R_{ab}.

Proof: Let $\mathcal{Z}$ be a minimal overlap strong optimal partition that satisfies all previous lemmas. Suppose (Z_a, Z_b) forms pattern 3, Z_a is an X-star and Z_b is a Y-star. We first show that any set, Z_c, that overlaps Z_a or Z_b, does so in pattern 1 such that Z_c is the trailer and overlaps one only of Z_a and Z_b. In addition there is at most one set, other than Z_b, that overlaps Z_a, and at most one set, other than Z_a, that overlaps Z_b.

Suppose Z_c overlaps Z_a. Z_c does not form pattern 2 or pattern 4 with Z_a, by lemma 5.9 and lemma 5.10 respectively.

If Z_c forms pattern 3 with Z_a then Z_c is a Y-star and Z_c overlaps Z_b. But Z_c can neither form pattern 3 nor pattern 1 with Z_b (respectively by lemma 5.7 and by the definition of pattern 1 since both are Y-stars). Since it cannot also form pattern 2 or pattern 4 with Z_b (by lemmas 5.9 and 5.10 resp.), therefore no set forms pattern 3 with either Z_a or Z_b.

Suppose Z_c crosses Z_a and $Z_c^Y < Z_a^Y$ (see figure 5.11 a). By lemma 5.11 Z_c is a Y-star, and $< Z_a \bigcup Z_c >$ is an L-Estar that contains x_p and y_q (lemma 5.3). But now Z_c overlaps Z_b in pattern 4 or crosses Z_b, both are contradictions (to lemma 5.8 and lemma 5.11, respectively).

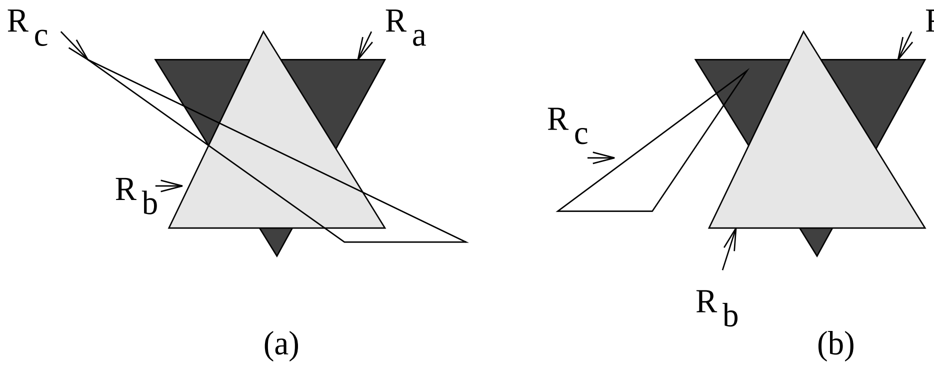

Figure 5.11: **(a)** Z_c **crosses** Z_a
(b) Z_c **forms pattern 1 with** Z_a

Therefore any set that overlaps either Z_a or Z_b forms pattern 1 with that set.

Suppose some set Z_c overlaps Z_a in pattern 1, Z_a is not the trailer since this would imply that Z_c overlaps Z_b and (Z_c, Z_b) does not form pattern 1. Therefore Z_c is the trailer and is a Y-star. By lemma 5.5 and since $R_b^X \not\subset R_c^X$ (otherwise there is a smaller partition by proposition 5.4) then Z_c does not overlap Z_b.

Now suppose Z_c is a Y-star such that $Z_c^X < Z_b^X$ (see figure 5.11 b).

If there is another trailer Z_d that overlaps Z_a, then Z_d is a Y-star that does not overlap Z_b. It cannot be true that $Z_d^X < Z_b^X$ (by corollary 5.2). Therefore if Z_d is a Y-star then $Z_d^X > Z_b^X$. But now $Z_b^Y \in R_a$, and all vertices in Z_b^X are either adjacent to the center of Z_c or to that of Z_d (since by convexity both centers are adjacent to s_a^X, one has neighbors smaller than Z_b^X and the other has neighbors larger than Z_b^X). Therefore by proposition 5.4 there is a smaller partition, which contradicts the minimality of $\mathcal{Z}$. Therefore only Z_c and Z_b overlap Z_a. A similar argument shows that there cannot be two trailers that overlap Z_b.

Let $Z_c' = Z_c \bigcup \{x : x \in Z_b, y < s_a^X\}$. If there is more than one vertex in Z_b larger than s_a^X then let $Z_b' = Z_b - Z_c'$ and $Z_a' = Z_a$, otherwise let $Z_a' = Z_a \bigcup Z_b^Y$ and $Z_b' = Z_b^Y - Z_c'$. Z_a' is a diameter two star and $R_a' = R_a$. Z_c' is a diameter two star that minimally overlaps Z_a' in pattern 1 (see proposition 5.8), and does not overlap Z_b'. Z_b' either has diameter zero or is a diameter two star that minimally overlaps Z_a'. Therefore $\mathcal{Z}' = \mathcal{Z} - \{Z_a, Z_b, Z_c\} \bigcup \{Z_a', Z_b', Z_c'\}$ is optimal.

We now show that $\mathcal{Z}'$ is a minimal overlap strong partition. Any set that overlaps or crosses one of Z_a', Z_b', or Z_c' overlaps or crosses at least one of Z_a, Z_b, and Z_c.

No set, other than Z_c and Z_b, overlaps Z_a.

Any set, Z_d, that overlaps Z_b is a diameter two X-star that forms pattern 1 with Z_b such that Z_d is the trailer, and Z_d neither crosses nor overlaps Z_a (by the argument in the above paragraph). If $s_d^X < s_a^X$ then Z_d crosses Z_c, minimally overlaps Z_c' in pattern 1, and overlaps neither Z_a' nor Z_b'. Otherwise $s_d^X > s_a^X$, Z_d does not overlap Z_c' or Z_a', and Z_d minimally overlaps Z_b' in pattern 1 or does not overlap Z_b' (if Z_b' has diameter zero).

Finally any set, Z_d, that overlaps or crosses Z_c and does not overlap Z_a or Z_b, must be such that Z_d crosses or minimally overlaps in pattern 1 Z_c, such that Z_d is the trailer. In both cases Z_d is a diameter two X-star and $Z_d^Y > Z_a^Y$ (by lemma 5.11 and lemma 5.5). In addition Z_d minimally overlaps in pattern 1 Z_c' such that Z_d is the trailer, and neither crosses nor overlaps Z_a' or Z_b'. Therefore $\mathcal{Z}'$ is a minimal overlap strong partition. This completes the proof for part (a).

We now prove part (b). Let Z_a and Z_b overlap in pattern 3 such that Z_a is an X-star while Z_b is a Y-star. Now suppose (s_a^Y, s_b^X) is not in E. Let x_i be in Z_c, where $c \neq a$ and $c \neq b$, $x_i < s_b^X$. Since no other set overlaps Z_a or Z_b (by part (a) of this lemma) and no edge has both endpoints smaller than or equal to (s_a^Y, s_b^X) (by lemma 4.2), therefore $|Z_c| = 1$. Suppose all neighbors of x_i are larger than R_{ab}, and let $y \in Z_d$, be one such neighbor. By proposition 5.2 $diam(Z_d - \{y\}) \leq diam(Z_d)$. But now $Z_a \bigcup Z_b \bigcup Z_c \bigcup \{y\}$ is an Estar with center in the set $\{s_a^X, s_b^Y, y\}$ (by biconvex S-ordering since (x_i, s_a^Y) is not in E and the edge (x_i, y) crosses both edges (s_b^Y, t_b^X) and (s_a^X, s_a^Y)), and by proposition 5.1 there is a smaller partition, which is a contradiction. Therefore every vertex smaller than R_{ab}^X has a neighbor in R_{ab}. Similarly every vertex smaller than R_{ab}^Y has a neighbor in R_{ab}.

If (t_a^Y, t_b^X) is not in E, then similarly we can prove that every vertex greater than R_{ab} has a neighbor in R_{ab}. ■

Let $\mathcal{Z}$ be a partition that satisfies all previous lemmas. We show in the next lemma that there is a minimal overlap strong optimal partition such that trailers are at the 'end' of the graph.

Lemma 5.13 (Pattern 1: Last lemma) *Let $\mathcal{Z}$ be a minimal overlap strong optimal partition that satisfies all previous lemmas. Then for any pair of sets, (Z_a, Z_b), in $\mathcal{Z}$ that forms pattern 1 such that Z_b is the trailer the following is true. Z_b is a Y-star that contains x_1 or x_p, or Z_b is an X-star that contains y_1 or y_q. Furthermore if Z_b contains x_1 or y_1 then (s_{ab}^Y, s_{ab}^X) is not in E, and if Z_b contains x_p or y_q then (t_{ab}^Y, t_{ab}^X) is not in E.*

Proof: Let $\mathcal{Z}$ be a minimal overlap strong optimal partition that satisfies all previous lemmas. Suppose Z_b is a Y-star and $Z_b^X < Z_a^X$.

The proof progresses in two steps. We first show that if the pair does not satisfy the lemma, then there is a non-overlapping subpartition of $Z_a \bigcup Z_b$ into sets Z_a' and Z_b' such that Z_b' is a diameter two star, and $diam(Z_a')$ is not larger than that of Z_a (step (a)). Therefore the partition $\mathcal{Z}' = \mathcal{Z} - \{Z_a, Z_b\} \bigcup \{Z_a', Z_b'\}$ is optimal. We finally show (in step (b)) that $\mathcal{Z}'$ is a minimal overlap strong partition that satisfies all previous lemmas and has fewer weak pairs, a contradiction to our choice of $\mathcal{Z}$.

Let x_1 be in Z_c. Assume $c \neq b$, or x_1 has a neighbor smaller than or equal to s_a^Y.

(a) We first prove that there is an edge with each endpoint smaller than or equal to s_b^X or s_a^Y. If $x_1 \in Z_b$ then this is true by our assumption.

Therefore suppose $c \neq b$. If x_1 has no neighbor smaller than or equal to s_a^Y, then any neighbor of x_1 is adjacent to all vertices in Z_b^X (by proposition 5.3 if that neighbor is in R_a or by biconvex S-ordering otherwise).

If $|Z_c| = 1$ then let $prev(x_1) \in Z_d$, $prev(x_1)$ is adjacent to all vertices in Z_b^X. By proposition 5.2 $diam(Z_d - \{prev(x_1)\}) \leq diam(Z_d)$. Let $Z_d' = Z_d - \{prev(x_1)\}$, $Z_b' = \{x_1, prev(x_1)\} \bigcup Z_b^X$, $Z_a' = Z_a \bigcup Z_b^Y$. Now $\mathcal{Z} - \{Z_a, Z_b, Z_c, Z_d\} \bigcup \{Z_a', Z_b', Z_d'\}$ is a smaller optimal partition which is a contradiction.

Therefore if $|Z_c| = 1$ then there is an edge with each endpoint smaller than or equal to s_b^X or s_a^Y.

Now suppose that $|Z_c| > 1$. All vertices in Z_b^X are adjacent to all neighbors of x_1 in Z_c. Therefore $diam(Z_c \bigcup Z_b^X) \leq diam(Z_c)$. Since $Z_b^Y \subset R_a$, then this implies there is a smaller optimal partition $\mathcal{Z} - \{Z_a, Z_b, Z_c\} \bigcup \{\{Z_a \bigcup Z_b^Y\}, \{Z_c \bigcup Z_b^X\}\}$.

Therefore there is an edge with each endpoint smaller than or equal to s_b^X or s_a^Y. Consequently s_a^Y is adjacent to all vertices in Z_b^X (by lemma 4.2 since the edges (s_b^X, t_b^Y) and $(s_a^Y, next(s_a^Y/Z_a))$ cross). Let $Z_a' = (Z_a \bigcup s_b^Y) - \{s_a^Y\}$, and $Z_b' = \{s_a^Y\} \bigcup Z_b^X$. Note that $diam(Z_b') = diam(Z_b)$, Z_a' and Z_b' do not overlap.

We now show that $diam(Z_a') \leq diam(Z_a)$. Note that $diam(Z_a' \bigcup \{s_a^Y\}) = diam(Z_a)$, by corollary 4.5. If $diam(Z_a') > diam(Z_a' \bigcup \{s_a^Y\})$, then by lemma 5.5 $diam(Z_a' \bigcup Z_b^X) = diam(Z_a \bigcup Z_b) \leq diam(Z_a)$ (since s_a^Y is adjacent to all vertices in Z_b^X). Therefore $diam(Z_a') \leq diam(Z_a)$, otherwise $\mathcal{Z}$ is not optimal.

Let $\mathcal{Z}' = \mathcal{Z} - \{Z_a, Z_b\} \bigcup \{Z_a', Z_b'\}$. $\mathcal{Z}'$ is an optimal partition.

(b) We now show that $\mathcal{Z}'$ is a minimal overlap strong partition that satisfies all lemmas. We will show that any weak pair in $\mathcal{Z}'$ minimally overlaps and corresponds to a weak pair in $\mathcal{Z}$. Since (Z_a', Z_b') is a strong pair, then $\mathcal{Z}'$ will have more strong pairs, a contradiction to the fact that $\mathcal{Z}$ is strong. First observe the following.

(i) No set forms pattern 2 or pattern 4 with either Z_a or Z_b (see lemma 5.9 and lemma 5.10).

(ii) No set, Z_c, crosses Z_a, since otherwise $< Z_a \bigcup Z_c >$ is an L-Estar or an R-Estar, and by lemma 4.5 $Z_a \bigcup Z_b \bigcup Z_c$ is an Estar, which is a contradiction to the minimality of $\mathcal{Z}$ by proposition 5.1.

(iii) By lemma 5.12 neither Z_a nor Z_b forms pattern 3 with any other set.

Now we consider all other ways a set Z_c can cross or overlap Z_a or Z_b.

Suppose Z_c overlaps Z_a.

If Z_c forms pattern 1 with Z_a and Z_a is the trailer, then Z_a is an X-star, Z_c does not overlap Z_b (by part (c) of lemma 5.5); Z_c minimally overlaps Z_a' in pattern 1, and neither crosses nor overlaps Z_b'. Suppose Z_c forms pattern 1 with Z_a and Z_c is the trailer. If Z_c is a

Y-star then by corollary 5.2 it cannot be true that $Z_c^X < Z_a^X$. Therefore if Z_c is the trailer then one of the following is true: Z_c is an X-star that crosses both Z_b and Z_b' and minimally overlaps in pattern 1 Z_a', Z_c neither crosses nor overlaps Z_b or Z_b' and minimally overlaps both Z_a and Z_a' in pattern 1.

Suppose now Z_c overlaps or crosses Z_b, but neither overlaps nor crosses Z_a.

If Z_c forms pattern 1 with Z_b, then Z_c is the trailer, and Z_c is a diameter two X-star that minimally overlaps Z_b' in pattern 1. Z_c neither crosses nor overlaps Z_a' in this case.

If Z_c crosses Z_b, then it is a diameter two X-star that crosses Z_b' and neither crosses nor overlaps Z_a'.

In all cases all pairs minimally overlap and satisfy all lemmas.

Therefore $\mathcal{Z}'$ is a minimal overlap strong optimal partition with fewer weak pairs, a contradiction to our choice of $\mathcal{Z}$. ∎

In what follows we assume that $\mathcal{Z}$ is a minimal overlap optimal partition that satisfies all preceding lemmas. We are now ready to prove the main theorem of this chapter. We show how, using all previous lemmas, we can express the optimal partition as the union of three disjoint subpartitions, one covering the center of the graph, $\mathcal{Z}_C$ (the center subpartition), and the other two each covering one end of the graph, $\mathcal{Z}_L$ and $\mathcal{Z}_R$ (the left and right subpartitions respectively). The left and right subpartitions will contain all trailers, set pairs that cross or form pattern 3, and enough sets so that each subpartition extends to one end of the graph.

Let $G = (X, Y, E)$ be a biconvex graph whose vertices, $X = \{x_1, \ldots, x_p\}$ and $Y = \{y_1, \ldots, y_q\}$, are biconvexly S-ordered.

Theorem 5.3 (Structure of Optimal Partition) *There is an optimal partition $\mathcal{Z}$, such that $\mathcal{Z}$ may be expressed as the union of three subpartitions $\mathcal{Z}_L$, $\mathcal{Z}_C$, and $\mathcal{Z}_R$, that are pairwise disjoint. No set in $\mathcal{Z}_C$ crosses any set in $\mathcal{Z}_L$ or $\mathcal{Z}_R$, and no set in $\mathcal{Z}_L$ overlaps or crosses any set in $\mathcal{Z}_R$. In addition, these subpartitions satisfy the following (see figure 5.12).*

1. *$\mathcal{Z}_C$ is either empty or is one of the following.*
   - *(a) $\mathcal{Z}_C$ is non-crossing and non-overlapping.*
   - *(b) $\mathcal{Z}_C = \mathcal{Z}$, and $\mathcal{Z}_C$ consists of a matching and cardinality one sets. Both (x_1, y_1) and (x_p, y_q) are not in E.*
   - *(c) $\mathcal{Z}_c = \mathcal{Z}$, and G has diameter three, four, or five. $\mathcal{Z}_C$ contains two diameter two stars, Z_i and Z_j, that overlap in pattern 3 and $< Z_i \bigcup Z_j >$ is an Estar in G. Every vertex, not in $Z_i \bigcup Z_j$, is smaller than R_{ij} or larger than R_{ij}, and is in a cardinality one set. Both (x_1, y_1) and (x_p, y_q) are not in E.*

2. *$\mathcal{Z}_L$ is either empty, or one of the following is true. Let G_L be the subgraph induced by all vertices in any set of $\mathcal{Z}_L$, and let x_R and y_R be the rightmost vertices in G_L. Then (x_R, y_R) is in E, and (x_1, y_1) is not in E.*

(a) G_L *contains both* x_1 *and* y_1. $\mathcal{Z}_L$ *consists of a matching and cardinality one sets. No set in* $\mathcal{Z}_L$ *in this case overlaps or crosses any set in* $\mathcal{Z}_C$.

(b) G_L *induces a subgraph of diameter three that contains both* x_1 *and* y_1. $\mathcal{Z}_L$ *consists of two diameter two stars that either cross or form pattern 1. One or both stars might be trailers with respect to some set in* $\mathcal{Z}_C$*; if both are trailers with respect to some set in* $\mathcal{Z}_C$*, then they cross.*

(c) G_L *induces a subgraph of diameter three or four that contains both* x_1 *and* y_1. $\mathcal{Z}_L$ *contains two diameter two stars,* Z_i *and* Z_j*, that overlap in pattern 3 and* $< Z_i \bigcup Z_j >$ *is an Estar in* G_L*. Every vertex in* G_L*, not in* $Z_i \bigcup Z_j$*, is smaller than* R_{ij} *and is in a cardinality one set. No set in* $\mathcal{Z}_L$ *in this case overlaps or crosses any set in* $\mathcal{Z}_C$.

(d) G_L *has diameter two.* $\mathcal{Z}_L$ *consists of a diameter two star that is a trailer with respect to some set in* $\mathcal{Z}_C$*, and contains* x_1 *or* y_1.

3. $\mathcal{Z}_R$ *is either empty, or one of the following is true. Let* G_R *be the subgraph induced by all vertices in any set of* $\mathcal{Z}_R$*, and let* x_L *and* y_L *be the leftmost vertices in* G_R*. Then* (x_L, y_L) *is in* E*, and* (x_p, y_q) *is not in* E.

(a) G_R *contains both* x_p *and* y_q. $\mathcal{Z}_R$ *consists of a matching and cardinality one sets. No set in* $\mathcal{Z}_R$ *in this case overlaps or crosses any set in* $\mathcal{Z}_C$.

(b) G_R *induces a subgraph of diameter three that contains both* x_p *and* y_q. $\mathcal{Z}_R$ *consists of two diameter two stars that either cross or form pattern 1. One or both stars might be trailers with respect to some set in* $\mathcal{Z}_C$*; if both are trailers with respect to some set in* $\mathcal{Z}_C$*, then they cross.*

(c) G_R *induces a subgraph of diameter three or four that contains both* x_p *and* y_q. $\mathcal{Z}_R$ *contains two diameter two stars,* Z_i *and* Z_j*, that overlap in pattern 3 and* $< Z_i \bigcup Z_j >$ *is an Estar in* G_R*. Every vertex in* G_R*, not in* $Z_i \bigcup Z_j$*, is larger than* R_{ij} *and is in a cardinality one set. No set in* $\mathcal{Z}_R$ *in this case overlaps or crosses any set in* $\mathcal{Z}_C$.

(d) G_R *has diameter two.* $\mathcal{Z}_R$ *consists of a diameter two star that is a trailer with respect to some set in* $\mathcal{Z}_C$*, and contains* x_p *or* y_q.

Proof : Choose a minimal overlap strong optimal partition $\mathcal{Z}$ that satisfies all previous lemmas. If $\mathcal{Z}$ is non-overlapping and non-crossing then let $\mathcal{Z}_C = \mathcal{Z}$, and we are done with the proof. Otherwise let $\mathcal{Z}_L = \mathcal{Z}_C = \mathcal{Z}_R = \phi$. Add to $\mathcal{Z}_C$ all sets Z_i such that Z_i neither crosses nor overlaps any other set. If $|Z_i| = 1$ then Z_i may later be moved from $\mathcal{Z}_C$ to $\mathcal{Z}_L$ or $\mathcal{Z}_R$.

Let Z_a and Z_b be two sets in $\mathcal{Z}$ that either overlap or cross. We consider all cases. In each case some set is appended to one of $\mathcal{Z}_R$, $\mathcal{Z}_C$, or $\mathcal{Z}_L$. Every time we add enough sets so that $\mathcal{Z}_L$ is either empty or contains one or both of x_1 and y_1, and $\mathcal{Z}_R$ is either empty or contains one or both of x_p and y_q. If $\mathcal{Z}_L$ (or $\mathcal{Z}_R$) contains exactly one of x_1 and y_1 (resp. x_p and y_q), then the other vertex is in $\mathcal{Z}_C$. Therefore at most one of these cases is true for each of $\mathcal{Z}_L$ and $\mathcal{Z}_R$.

After we consider all cases, we show that no set in $\mathcal{Z}_L$ can overlap or cross any set in $\mathcal{Z}_R$, and describe how sets in $\mathcal{Z}_C$ might overlap or cross sets in $\mathcal{Z}_L$ or $\mathcal{Z}_R$.

Since $\mathcal{Z}$ satisfies all lemmas, we know that (Z_a, Z_b) does not form pattern 2 or pattern 4 (by lemma 5.9 and lemma 5.10), and we have one of the following three cases.

Case 1: Z_a and Z_b cross.

Then by lemma 5.4 $Z_a \bigcup Z_b$ induces an R-Estar (or an L-Estar). As well either both are diameter two stars (by lemma 5.11) and both x_1 and y_1 (resp. x_p and y_q) are in $Z_a \bigcup Z_b$, or that subpartition is a matching and both x_1 and y_1 (resp. x_p and y_q) are in cardinality one or cardinality two sets (corollary 5.1).

If $Z_a \bigcup Z_b$ induces an R-Estar and both sets have diameter two, we add Z_a and Z_b to $\mathcal{Z}_L$. If they induce an L-Estar and both sets have diameter two, we add Z_a and Z_b to $\mathcal{Z}_R$. In this case $\mathcal{Z}_L$ and $\mathcal{Z}_R$ satisfy part (b) of the theorem.

Now consider the case where both have diameter one. Let G_a be the subgraph induced by all crossing set pairs that have diameter one. Let x_L and y_L be the smallest index vertices in G_a, and x_R and y_R be the largest index vertices in G_a.

If (x_R, y_R) is in E, then all crossing set pairs of diameter one, in $\mathcal{Z}$, induce R-Estars (by lemma 5.4 and lemma 4.2). We add all sets with vertices in or smaller than G_a to $\mathcal{Z}_L$. Note that each of x_1 and y_1 is in a cardinality one or cardinality two set, that is in $\mathcal{Z}_L$ (see corollary 5.1), and (x_1, y_1) is not in E.

If (x_L, y_L) is in E, then we add all sets with vertices in or larger than G_a to $\mathcal{Z}_R$.

Otherwise both (x_R, y_R) and (x_L, y_L) are not edges in E. In this case it must be true that there is a pair of sets that cross and induce an R-Estar, a pair of sets that cross and induce an L-Estar, all are diameter one sets, and each of x_1, y_1, x_p, and y_q is either in a diameter one set or in a diameter zero set (see corollary 5.1).

If two diameter one sets, Z_i and Z_j, cross such that Z_i crosses some other set Z_{i1}, and Z_j crosses some other set Z_{j1}, and $Z_i \bigcup Z_{i1}$ induces an R-Estar while $Z_j \bigcup Z_{j1}$ induces an L-Estar, then all sets in $\mathcal{Z}$ have diameter one or diameter zero (see corollary 5.1, since all vertices in G are in $R(Z_i \bigcup Z_j)$, in or smaller than $R(Z_i \bigcup Z_{i1})$, or in or larger than $R(Z_j \bigcup Z_{j1})$). In this case we add all sets to $\mathcal{Z}_C$. $\mathcal{Z}_C$ satisfies part (b) in this case.

Otherwise there are largest index vertices x_{L1} and y_{L1} in G_a and smallest index vertices x_{R1} and y_{R1} in G_a such that no set with vertices in the set $S_L = \{x, y : x \leq x_{L1}, y \leq y_{L1}\}$

crosses any set with vertices in the set $S_R = \{x, y : x \geq x_{R1}, y \geq y_{R1}\}$, all crossing set pairs with vertices in S_L induce an R-Estar, all crossing set pairs with vertices in S_R induce an L-Estar, $(x_{L1}, y_{L1}) \in E$, and $(x_{R1}, y_{R1}) \in E$. Let $\mathcal{Z}_L$ contain all sets with vertices in S_L, while $\mathcal{Z}_R$ contain all sets with vertices in S_R.

Each of $\mathcal{Z}_L$ and $\mathcal{Z}_R$ satisfies part (a) of the theorem.

Case 2: Z_a and Z_b form pattern 3.

In this case both are diameter two stars and induce an Estar (lemma 5.7).

If both (s^Y_{ab}, s^X_{ab}) and (t^Y_{ab}, t^X_{ab}) are not in E then we add Z_a, Z_b, and all other sets to $\mathcal{Z}_C$. By lemma 5.12 and the fact that no edge has both endpoints smaller than or larger than R_{ab}, if $|\mathcal{Z}| > 2$ then any other set in $\mathcal{Z}$ has cardinality one. By lemma 5.12 any vertex not in R_{ab} has a neighbor in R_{ab} (by lemma 5.12). It follows that G has diameter at least three and at most five, $\mathcal{Z}_C = \mathcal{Z}$, and both (x_1, y_1) and (x_p, y_q) are not in E.

$\mathcal{Z}_C$ satisfies part (c) of the theorem.

Otherwise exactly one of (s^Y_{ab}, s^X_{ab}) or (t^Y_{ab}, t^X_{ab}) is not in E (see lemma 5.7). If (t^Y_{ab}, t^X_{ab}) is not in E then we add Z_a, Z_b, and all sets that contain vertices larger than R_{ab} to $\mathcal{Z}_R$. Otherwise we add Z_a, Z_b, and all sets that contain vertices smaller than R_{ab} to $\mathcal{Z}_L$. Suppose Z_a and Z_b have been added to $\mathcal{Z}_R$. Then $(t^Y_{ab}, t^X_{ab}) = (x_L, y_L) \in E$, and (x_p, y_q) is not in E (by our choice of $\mathcal{Z}_R$). Let Z_c be a set that contains a vertex larger than R_{ab}. Then by lemma 5.12 Z_c has cardinality one, and has a neighbor in R_{ab}. Therefore the diameter of G_R is smaller than or equal to four, and (x_p, y_q) is not in E.

$\mathcal{Z}_L$ and $\mathcal{Z}_R$ satisfy part (b) or part (c) of the theorem.

Case 3: Z_a and Z_b form pattern 1.

Suppose at least one of Z_a and Z_b is not in $\mathcal{Z}_C$, $\mathcal{Z}_L$, or $\mathcal{Z}_R$, and Z_b is the trailer with respect to Z_a. If Z_b contains one of x_1 or y_1, we add Z_b to $\mathcal{Z}_L$. Otherwise it contains one of x_p or y_q (see lemma 5.13), and in this case we add Z_b to $\mathcal{Z}_R$. Suppose Z_b is in $\mathcal{Z}_L$, and is an X-star that contains y_1. By lemma 5.13 (x_1, y_1) is not in E.

If x_1 is in a cardinality one set, then it is in a set that is already in $\mathcal{Z}_C$.

Suppose x_1 is not in a cardinality one set.

If x_1 is not in Z_a then let x_1 be in Z_c. One of the following is true: Z_c crosses Z_b and both sets are already in $\mathcal{Z}_L$, Z_c forms pattern 1 with Z_b such that it is the trailer, we add Z_c to $\mathcal{Z}_L$. Note that t^X_b and t^Y_c (or t^X_b and t^Y_b) are the largest index vertices in G_L, and are adjacent.

Otherwise suppose x_1 is in Z_a. If Z_a does not overlap any other set or overlaps some set in pattern 1 but is not a trailer, then we add Z_a to $\mathcal{Z}_C$. Notice that Z_a does not cross any set, Z_c, otherwise $Z_a \bigcup Z_c$ induces an L-Estar or an R-Estar, by lemma 4.5 $Z_a \bigcup Z_b \bigcup Z_c$ is an Estar, and by proposition 5.1 there is a smaller partition.

By lemma 5.12 Z_a cannot form pattern 3 with any set.

If Z_a is a trailer with respect to some set then it is a Y-star. We add both Z_a and Z_b to $\mathcal{Z}_L$. Note that $< Z_a \bigcup Z_b >$ is an Estar, and that s_a^Y and s_a^X are the largest index vertices in the subgraph induced by $\mathcal{Z}_L$ in this case and are adjacent.

$\mathcal{Z}_L$ and $\mathcal{Z}_R$ satisfy part (b) or part (d) of the theorem.

By all the above cases it is clear that $\mathcal{Z}_C \bigcup \mathcal{Z}_L \bigcup \mathcal{Z}_R = \mathcal{Z}$, and the three subpartitions are mutually disjoint.

We now show that no set in $\mathcal{Z}_L$ can overlap or cross any set in $\mathcal{Z}_R$, and describe the cases where sets in $\mathcal{Z}_L$ (or $\mathcal{Z}_R$) might overlap or cross sets in $\mathcal{Z}_C$.

Suppose first that $\mathcal{Z}_L$ consists of a matching and cardinality one sets. No set in $\mathcal{Z}_L$ overlaps any set, otherwise $\mathcal{Z}$ is not a minimal overlap partition. No set in $\mathcal{Z}_L$ crosses any set outside of $\mathcal{Z}_L$ (by our choice of $\mathcal{Z}_L$, $\mathcal{Z}_R$, and $\mathcal{Z}_C$).

If $\mathcal{Z}_L$ contains two sets that overlap in pattern 3, then by our choice of $\mathcal{Z}_L$ and by lemma 5.12, no set in $\mathcal{Z}_L$ overlaps or crosses any set outside of $\mathcal{Z}_L$.

Suppose now $\mathcal{Z}_L$ contains a diameter two star, Z_i, and Z_i crosses or overlaps some set Z_l, not in $\mathcal{Z}_L$, in a pattern other than pattern 3.

Suppose Z_l is in $\mathcal{Z}_R$. Assume that Z_i is an X-star. Z_i contains y_1 (since Z_i is in $\mathcal{Z}_L$). Since Z_l is in $\mathcal{Z}_R$ then Z_l is a star. Z_i cannot cross Z_l otherwise they are both in $\mathcal{Z}_L$ or both in $\mathcal{Z}_R$. Therefore Z_i and Z_l form pattern 1. Suppose Z_i is the trailer with respect to Z_l. Then Z_l is a Y-star, and since it is in $\mathcal{Z}_R$ then it contains x_p. Z_l also is a trailer with respect to some set Z_j, which implies that Z_j crosses Z_i. This implies that Z_j is a Y-star by lemma 5.11, and thus cannot form pattern 1 with Z_l, a contradiction. Therefore no set in $\mathcal{Z}_L$ overlaps or crosses any set in $\mathcal{Z}_R$.

Now let Z_i be in $\mathcal{Z}_L$ and Z_l be in $\mathcal{Z}_C$. Therefore $\mathcal{Z}_L$ in this case either consists entirely of Z_i, or contains another Y-star Z_j; Z_j forms pattern 1 with Z_i and does not overlap any other set (follows by lemma 5.5 and lemma 5.12), or Z_j crosses Z_i and might overlap some set in $\mathcal{Z}_C$ in pattern 1. ■

Figure 5.12 depicts all cases for the subpartitions $\mathcal{Z}_R$, $\mathcal{Z}_C$, and $\mathcal{Z}_L$ (ignoring some symmetric cases for $\mathcal{Z}_L$ and $\mathcal{Z}_R$).

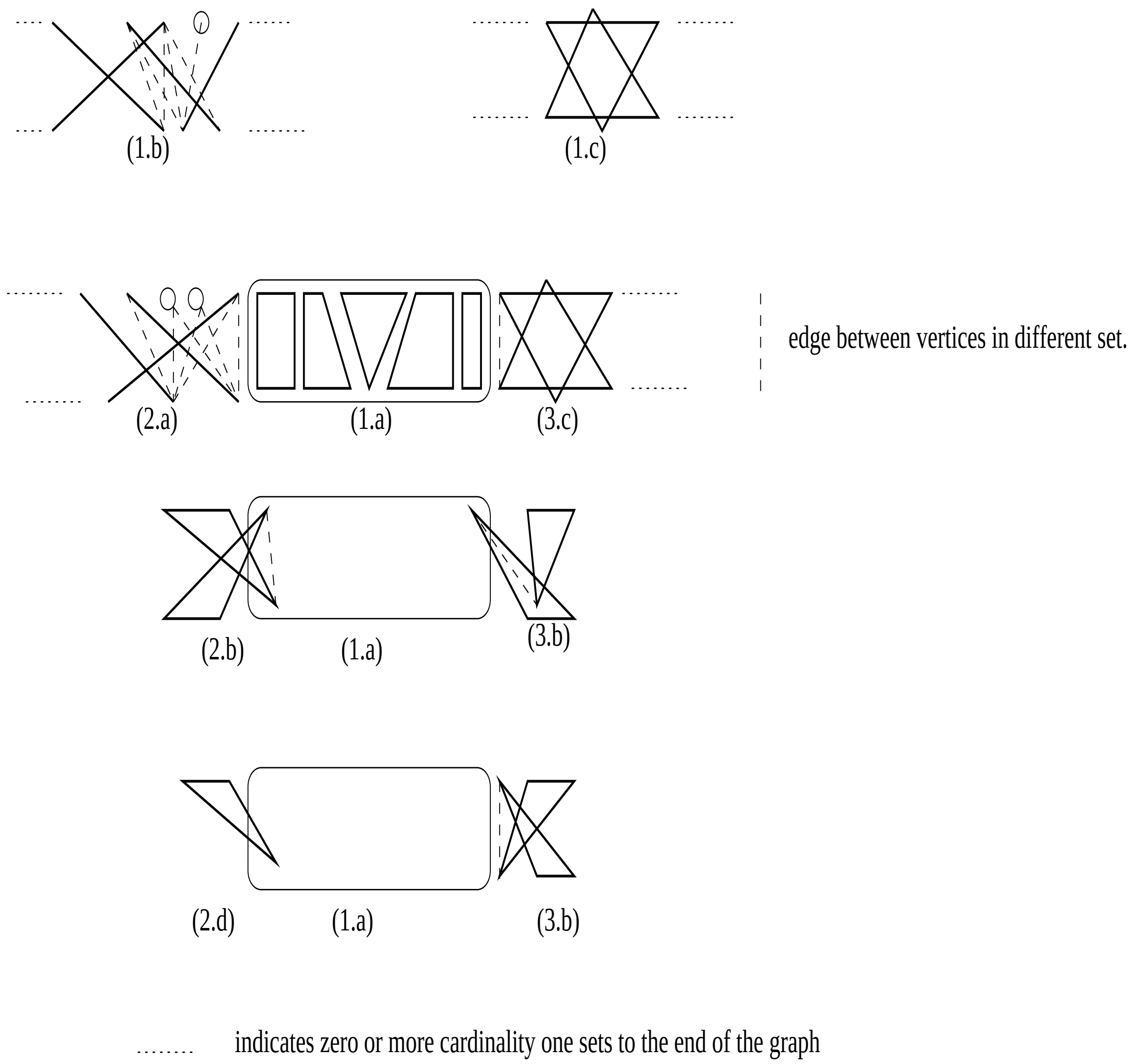

Figure 5.12: Some of the Possibilities for $\mathcal{Z}_L$, $\mathcal{Z}_C$, and $\mathcal{Z}_R$ Labels correspond to items in theorem 5.3

Corollary 5.3 (Bipartite Permutation Partition) *If G is a bipartite permutation graph then G has an optimal non-crossing non-overlapping partition.*

Proof : We can see this by reviewing each overlap pattern in the partition $\mathcal{Z}$ that satisfies all lemmas. By lemma 5.5 and lemma 5.13 if (Z_a, Z_b) form pattern 1 then they do not induce a strongly ordered BPG. The same is true if (Z_a, Z_b) form pattern 3 (by lemma 5.10), or cross (by lemma 5.2). ∎

Chapter 6

Partitioning Biconvex Graphs: Algorithms

6.1 Preliminaries

We concluded the previous chapter by showing how to obtain an optimal partition, $\mathcal{Z}$, for problem $PVMF$ (Partitioning Vertices to Minimize Monotone Function) on biconvex graphs as the union of three disjoint subpartitions $\mathcal{Z}_L$, $\mathcal{Z}_C$, and $\mathcal{Z}_R$ (theorem 5.3). We call these subpartitions *the left subpartition* ($\mathcal{Z}_L$), *the center subpartition* ($\mathcal{Z}_C$), and *the right subpartition* ($\mathcal{Z}_R$). In this chapter we show how these results may be used to obtain an efficient parallel algorithm for $PVMF$ on biconvex graphs, and an efficient sequential algorithm for PVD (Partitioning Vertices to Minimize Maximum Diameter) on bipartite permutation graphs.

In the next section we show how all possibilities for the cardinality of $\mathcal{Z}_C$ may be obtained efficiently in parallel. Computing $\mathcal{Z}_R$ is similar to computing $\mathcal{Z}_L$, due to the symmetry between them. In the third section of this chapter we show how to compute the cardinality of $\mathcal{Z}_L$ (and $\mathcal{Z}_R$) efficiently in parallel. In the fourth section we state the parallel algorithm. In the final section we present the sequential algorithm on bipartite permutation graphs.

Before we start the next section we present two lemmas that describe algorithms we will need throughout the next two sections. The first lemma describes a well known technique to efficiently compute in parallel the value of the objective function, $F = FM(1) \bullet FM(2) \ldots FM(\alpha)$, for a given array of integers FM, where α is a power of two. In the second lemma we present an algorithm to compute all possible values for the cardinality and the objective function for a partition that consists entirely of cardinality two and cardinality one sets, efficiently in parallel.

Let $G = (X, Y, E)$ be a biconvex graph whose vertices have a biconvex S-ordering. Let $n = |X| + |Y|$ and $m = |E|$.

Lemma 6.1 (Computing Objective Function) *Given an array of integers FM, of dimension α, α is a power of two, the value $F = FM(1) \bullet FM(2) \bullet \ldots FM(\alpha)$, may be computed efficiently in parallel in $\mathcal{O}(\log \alpha)$ time using $\mathcal{O}(\alpha)$ processors.*

Proof: We use the following procedure that primarily relies on the associativity of $\bullet$. It is merely an implementation of the well known balanced binary tree technique (see [36]).

Procedure OBJ-FUN(α, FM, F)

begin

$inc = 2$

repeat $\log \alpha$ **times**

  for all $i \in \{1, 1 + inc, 1 + 2 * inc, \ldots\}$ **do in parallel**

    $FM(i) = FM(i) \bullet FM(i + inc/2)$

  enddo

$inc = 2 * inc$

$F = FM(1)$

end ∎

Lemma 6.2 (All Matchings) *Let $\mathcal{Z}$ be a partition for a biconvex graph, $G = (X, Y, E)$, that consists of a matching and cardinality one sets. There are $\mathcal{O}(m)$ values for the cardinality and the value of the objective function of such a partition, that may be computed efficiently in parallel in $\mathcal{O}(\log^2 n)$ time using $\mathcal{O}(mn)$ processors.*

Proof: Any such partition may be expressed as the union of two subpartitions $\mathcal{Z}_1$ and $\mathcal{Z}_2$, where $\mathcal{Z}_1$ consists entirely of cardinality one sets and $\mathcal{Z}_2$ consists entirely of cardinality two sets. Moreover $|\mathcal{Z}_2| = CARD2 - j$, for some j, $0 \leq j \leq CARD2$, where $CARD2$ is the cardinality of a maximum matching for G. Now let $CARD1 = n - 2 * CARD2$, then $|\mathcal{Z}_1| = CARD1 + 2 * j$. Therefore there are $CARD2 = \mathcal{O}(m)$ values for the cardinality of such a partition for any graph G.

We will first find the cardinality of a maximum matching for G. For that we will use the parallel algorithm given in [24], developed by Dekel and Sahni. Their algorithm requires $\mathcal{O}(\log^2 n)$ time and $\mathcal{O}(n)$ processors. Suppose the output of the algorithm is, $CARD2$, the cardinality of a maximum matching. Then $CARD1 = n - 2 * CARD2$.

Let $CARD2(j) = CARD2 - j$; i.e. $CARD2(j)$ is the number of cardinality two sets remaining after removing j edges from a maximum matching of size $CARD2$. Let $CARD1(j) = CARD1 + 2j$.

Therefore to obtain all possible values of the objective function and the cardinality for such a partition, we perform the following procedure. The input to the procedure is the graph G, and the output consists of two arrays FT and K, each of dimension m, where m is the number of edges in G. $FT(j)$ holds the value of the objective function for a partition that consists entirely of cardinality one and cardinality two sets and contains exactly $CARD2 - j$ cardinality two sets. $K(j)$ holds the corresponding cardinality for that partition.

Consider Δ to be the *identity*, i.e. $\Delta \bullet x = x$.

Procedure ALL-MATCHINGS(G, FT, K)

begin

1. Compute $CARD2$ and $CARD1$ using the algorithm in [24].
2. For all j between zero and $CARD2$, do the following (in parallel).
   (a) Let $\alpha = 2^{\lceil \log(CARD2(j)+CARD1(j)) \rceil}$.
   (b) Let $FM(i) = 1$ if $i \le CARD2(j)$, $FM(i) = 0$ if $CARD2(j) < i \le CARD1(j) + CARD2(j)$, and all other values of $FM(i)$ are equal to Δ, where $i \le \alpha$. This may also be done in parallel.
   (c) Call OBJ-FUN(α, FM, F).
   (d) Let $FT(j) = F$ and $K(j) = CARD2(j) + CARD1(j)$

end

To compute $CARD2$ and $CARD1$ we need $\mathcal{O}(\log^2 n)$ time and $\mathcal{O}(n)$ processors. We need m processors for each value of j, and n processors to calculate FM. Calculating F requires $\mathcal{O}(\log n)$ time and $\mathcal{O}(n)$ processors. Therefore in total we need $\mathcal{O}(\log^2 n)$ time and $\mathcal{O}(mn)$ processors. ■

6.2 Computing The Center Subpartition $\mathcal{Z}_C$

We start this section by presenting a parallel algorithm PVMF-CENTER that finds the cardinality of an optimal non-crossing non-overlapping partition, which is one possibility for $\mathcal{Z}_C$ (recall theorem 5.3). We will use that algorithm later as part of a larger one to compute an optimal partition for a biconvex graph G. The algorithm is similar to algorithm 3.1 in chapter 3.

We need the following definitions.

6.2.1 Definitions and Overview

Let $\mathcal{Z}$ be an optimal non-crossing non-overlapping partition.

Let $\mathcal{Z}_{a,b}^{c,d} = \mathcal{Z}(a,b,c,d)$ be an optimal non-crossing non-overlapping partition for the set of vertices $R(\mathcal{Z}_{a,b}^{c,d}) = \{x : x_a \le x \le x_c\} \bigcup \{y : y_b \le y \le y_d\}$.

Let $Q^*(a,b,c,d)$ be the cardinality of $\mathcal{Z}_{a,b}^{c,d}$, and $F(a,b,c,d)$ be the value of the objective function for that partition.

We start the algorithm by initially setting $F(a,b,c,d)$ to the diameter of the set $R(\mathcal{Z}_{a,b}^{c,d})$. $Q^*(a,b,c,d)$ is initially set to one if the corresponding value of F is less than or equal to B.

Suppose now all values of Q^* that are less than or equal to l have already been computed. And suppose that $\mathcal{Z}_{a,b}^{c,d}$ is an optimal partition of cardinality l, for some x_a, y_b, x_c, and y_d. Then, as we will prove shortly, there are vertices x_i, and y_j, where $x_a \leq x_i \leq x_c$ and $y_b \leq y_j \leq y_d$ such that $F(a,b,i,j) \bullet F(i+1,j+1,c,d)$ is less than or equal to B, and $Q^*(a,b,c,d) = Q^*(a,b,i,j) + Q^*(i+1,j+1,c,d)$.

Before presenting the algorithm, we list a set of assumptions, and present a theorem that will translate directly to a parallel algorithm.

Assumption 1: The vertices v_Λ and v_Υ are the *null* vertices.

Assumption 2: When considering a set $\{x_{i1},\ldots,x_{i2},y_{j1},\ldots,y_{j2}\}$, we assume that each of x_{i1} and y_{j1} is preceded by v_Λ , and that each of x_{i2} and y_{j2} is succeeded by v_Υ, i.e. $\Lambda+1 = x_{i1}$ or $\Lambda+1 = y_{j1}$, $x_{i2}+1 = \Upsilon$, and $y_{j2}+1 = \Upsilon$.

Assumption 3: We will use Λ and Υ to be able to account for subpartitions that consists entirely of cardinality one sets in X or consist entirely of cardinality one sets in Y. To clarify, let us suppose $\mathcal{Z}_{a,b}^{c,d} = \mathcal{Z}_1 \bigcup \mathcal{Z}_2$ is a partition for a graph G, where $\mathcal{Z}_1$ and $\mathcal{Z}_2$ neither cross nor overlap, and each of $\mathcal{Z}_1$ and $\mathcal{Z}_2$ consists of consecutive vertices in X and consecutive vertices in Y. Suppose $\mathcal{Z}_2$ consist entirely of cardinality one sets in X, and $\mathcal{Z}_1$ does not consist entirely of cardinality one sets in X. Assume $\mathcal{Z}_2 = \{\{x_{i1}\},\ldots\{x_{i2}\}\}$. Then $x_{i1} = x_a$ or $x_{i2} = x_c$ (otherwise $\mathcal{Z}_1$ does not consist of consecutive vertices in X). If $x_{i1} = x_a$ then we will represent $\mathcal{Z}_2$ as $\mathcal{Z}(a,b,i2,\Lambda)$, and in this case $\mathcal{Z}_{a,b}^{c,d} = \mathcal{Z}(a,b,i2,\Lambda) \bigcup \mathcal{Z}(i2+1,b,c,d)$ (see assumption 2). Otherwise $x_{i2} = x_c$ then we will represent $\mathcal{Z}_2$ as $\mathcal{Z}(i1,\Upsilon,c,d)$, and in this case $\mathcal{Z}_{a,b}^{c,d} = \mathcal{Z}(a,b,i1-1,d) \bigcup \mathcal{Z}(i1,\Upsilon,c,d)$ (see assumption 2).

Therefore when initially calculating the values of F and Q^*, we assume that $F(i1,j1,i2,\Lambda) = F(i1,\Upsilon,i2,j2) =$ the value of the objective function for the subpartition that consists of the cardinality one sets $\{x_{i1}\},\{x_{i1+1}\}\ldots\{x_{i2}\}$, and $F(i1,j1,\Lambda,j2) = F(\Upsilon,j1,i2,j2) =$ the value of the objective function for the subpartition that consists of the cardinality one sets $\{y_{j1}\},\{y_{j1+1}\}\ldots\{y_{j2}\}$. The values of Q^* are computed accordingly. All other values of $F(i1,j1,i2,j2)$, where one or more of the arguments $i1,i2,j1,j2$ is Λ or Υ, are assumed equal to ∞.

6.2.2 A Parallel Algorithm for a non-crossing non-overlapping partition

The next theorem will enable us to devise an efficient parallel algorithm.

Theorem 6.1

$$Q^*(a,b,c,d) = \min_{\substack{a \le i \le c,\, or\ i = \Lambda \\ b \le j \le d,\, or\ j = \Lambda \\ f(\mathcal{Z}_{a,b}^{i,j}) \bullet f(\mathcal{Z}_{i+1,j+1}^{c,d}) \le B}} (Q^*(a,b,i,j) + Q^*(i+1,j+1,c,d)),$$

where each of $Q^*(a,b,i,j)$ and $Q^*(i+1,j+1,c,d)$ is not larger than $\lceil (1/2) * Q^*(a,b,c,d) \rceil$.

Proof: Let $|\mathcal{Z}| = Q^*(a,b,c,d)$. We prove the claim by induction on $Q^*(a,b,c,d)$.

The hypothesis is true for $Q^*(a,b,c,d) = 1$ since the cardinality of any optimal partition in this case will equal $Q^*(a,b,c,d)$ for some a, b, c, and d. If the optimal partition is a cardinality one set then $d = \Lambda$ and $c = a$, or $a = \Upsilon$ and $b = d$ will give the desired result (see assumption 2). It is also true for $Q^*(a,b,c,d) = 2$ since $\mathcal{Z}$ is non-crossing and non-overlapping.

Assume it is true for all biconvex graphs that have an optimal partition of cardinality smaller than k. Let $G = (X,Y,E), X \bigcup Y = \{x_a, \ldots, x_c, y_b, \ldots, y_d\}$, be a biconvex graph that has a cardinality k optimal non-crossing non-overlapping partition $\mathcal{Z} = \{Z_1, Z_2, \ldots, Z_{k-1}, Z_k\}$.

$f(\mathcal{Z}) = diam(Z_1) \bullet diam(Z_2) \bullet \ldots \bullet diam(Z_r) \ldots\ldots\ldots \bullet diam(Z_k)$

$= (diam(Z_1) \bullet diam(Z_2) \ldots \bullet diam(Z_r)) \bullet (diam(Z_{r+1}) \ldots \bullet diam(Z_k)) \le B$ (since $\bullet$ is associative), where $|Z_i| \ge 1$ for all i, $1 \le i \le k$.

Therefore there are two disjoint subsets of $\mathcal{Z}$, $\mathcal{Z}_1$ and $\mathcal{Z}_2$, such that $\mathcal{Z}_1 \bigcup \mathcal{Z}_2 = \mathcal{Z}$, $\mathcal{Z}_1 = \{Z_1, Z_2, \ldots, Z_r\}$, $\mathcal{Z}_2 = \{Z_{r+1}, Z_{r+2}, \ldots, Z_k\}$, and $f(\mathcal{Z}_1) \bullet f(\mathcal{Z}_2) \le B$.

Since $1 \le |\mathcal{Z}_1| \le k-1$, $1 \le |\mathcal{Z}_2| \le k-1$, therefore $Q^*(a,b,c,d) = Q^*(a,b,t_r^X,t_r^Y) + Q^*(t_r^X+1, t_r^Y+1, c, d)$, where $t_r^X + 1 = s_{r+1}^X$ and $t_r^Y + 1 = s_{r+1}^Y$ since no edges from different sets cross, and $a \le t_r^X \le c$ or $t_r^X = \Lambda$, and $b \le t_r^Y \le d$ or $t_r^Y = \Lambda$. For instance $t_r^X = \Lambda$ gives $Q^*(a,b,c,d) = Q^*(a,b,\Lambda,t_r^Y) + Q^*(a, t_r^Y+1, c, d)$, and $t_r^X = c$ gives $Q^*(a,b,c,d) = Q^*(a,b,c,t_r^Y) + Q^*(\Upsilon, t_r^Y+1, c, d)$, see assumptions 2 and 3 for interpretations. Moreover, by the associativity of $\bullet$, there are such subpartitions $\mathcal{Z}_1$ and $\mathcal{Z}_2$, that satisfy $|\mathcal{Z}_1|, |\mathcal{Z}_2| \le \lceil 1/2 * |\mathcal{Z}| \rceil$.

Since both $Q^*(a,b,t_r^X,t_r^Y)$ and $Q^*(t_r^X+1, t_r^Y+1, c, d)$ are correctly computed by the induction hypothesis, then $Q^*(a,b,c,d) = k$ is correctly computed. ■

Algorithm 6.1 Procedure PVMF-CENTER(G_1, B, k, F)

Input: Integer B and $G_1(X,Y,E)$ where G_1 is a biconvex graph whose vertices have a biconvex S-ordering, $X = \{x_1, \ldots, x_{|X|}\}$, $Y = \{y_1, \ldots, y_{|Y|}\}$, $n = |X| + |Y|$.
Output: The cardinality k of an optimal partition.

begin

1. **for all** x_{i1} *and* x_{i2} *in* $X \bigcup \{\Lambda, \Upsilon\}$, y_{j1} *and* y_{j2} *in* $Y \bigcup \{\Lambda, \Upsilon\}$, $x_{i1} \le x_{i2}$, $y_{j1} \le y_{j2}$ **do in parallel**

$Q^*(i1, j1, i2, j2) = \infty$

/Assumption 3 applies here/

$F(i1, j1, i2, j2) = diam(\{x_{i1}, x_{i1+1}, \ldots, x_{i2}, y_{j1}, \ldots, y_{j2}\})$

if $F(i1, j1, i2, j2) \leq B$ **then** $Q^*(i1, j1, i2, j2) = 1$

endfor

2. **repeat** $\lceil \log n \rceil$ *times*

(a) **for all** $i1, i2, l1, l2, j1, j2$ *where* $i1 \leq l1 \leq i2$ *or* $l1 = \Lambda$ *, and* $j1 \leq l2 \leq j2$ *or* $l2 = \Lambda$ **do in parallel**

$W[i1, j1, l1, l2, i2, j2] = \infty$

if $(F(i1, j1, l1, l2) \bullet F(l1 + 1, l2 + 1, i2, j2) \leq B)$

/Assumption 2 applies here/

then $W[i1, j1, l1, l2, i2, j2] = Q^*(i1, j1, l1, l2) + Q^*(l1 + 1, l2 + 1, i2, j2)$

endif

endfor

(b) **for all** $i1, i2, l1, l2, j1, j2$ *where* $i1 \leq l1 \leq i2$ *or* $l1 = \Lambda$ *, and* $j1 \leq l2 \leq j2$ *or* $l2 = \Lambda$ **do in parallel**

$Q^*(i1, j1, i2, j2) = \min(\min_{l1,l2} W[i1, j1, l1, l2, i2, j2],$
$Q^*(i1, j1, i2, j2))$

if $Q^*(i1, j1, i2, j2) \neq W[i1, j1, l1, l2, i2, j2]$ *for some l1,l2*

then $l1^* = 0$ *and* $l2^* = 0$

else $l1^* = \{l1 : Q^*(i1, j1, i2, j2) = W[i1, i2, l1, l2, j1, j2]\}$

and $l2^* = \{l2 : Q^*(i1, j1, i2, j2) = W[i1, i2, l1, l2, j1, j2]\}$

endif

if $l1^* \neq 0$ *and* $l2^* \neq 0$

then $F(i1, j1, i2, j2) = F(i1, j1, l1^*, l2^*) \bullet F(l1^* + 1, l2^* + 1, i2, j2)$

endif

endfor

3. $k = Q^*(1, 1, |X|, |Y|)$

4. $F = F(1, 1, |X|, |Y|)$

end

Theorem 6.2 *Algorithm 6.1 finds the cardinality of an optimal non-crossing non-overlapping partition in* $\mathcal{O}(\log^2 n)$ *time using* $\mathcal{O}(n^8)$ *processors.*

Proof : Proof of correctness follows directly from theorem 6.1.

The first loop in the algorithm requires n^4 processors, one for each $x_{i1}, y_{j1}, x_{i2}, y_{j2}$. Another n^2 processors and $\mathcal{O}(\log n)$ time are needed to calculate the diameter (second statement in that loop) by lemma 4.10.

The next two loops, 2.a and 2.b, are repeated $\mathcal{O}(\log n)$ times. The second of these loops requires more resources. For each $i1, i2, l1, l2, j1, j2$ a total of n^6 processors are needed. The first statement in loop 2.b requires $\mathcal{O}(n^2)$ processors and $\mathcal{O}(\log n)$ time for the min operation. ■

Corollary 6.1 *The cardinality of the optimal partition for a bipartite permutation graph may be computed in $\mathcal{O}(\log^2 n)$ time using $\mathcal{O}(n^8)$ processors.*

6.2.3 Computing All Possible Values of a Partition $\mathcal{Z}_C$

Let $G = (X, Y, E)$ be a biconvex graph. Let $\mathcal{Z}$ be an optimal partition that satisfies theorem 5.3. Let $\mathcal{Z}_C$ be the subpartition of $\mathcal{Z}$ that satisfies one of the conditions 1.a, 1.b, or 1.c in the statement of that theorem. Suppose we are given G_C, the subgraph of G that is induced by all sets in $\mathcal{Z}_C$, and G_C is non-empty.

Theorem 6.3 *There are $\mathcal{O}(m)$ possible values for the cardinality and the objective function for an optimal partition $\mathcal{Z}_C$ that may be calculated in parallel using $\mathcal{O}(Bn^8)$ processors and $\mathcal{O}(\log^2 n)$ time.*

Proof : Let F_C be an array that holds values of the objective function for all possible values of $\mathcal{Z}_C$, i.e. any partition that satisfies one of the conditions 1.a, 1.b, or 1.c in theorem 5.3. Let the array K_C hold the corresponding value for the cardinality of $\mathcal{Z}_C$. Referring to theorem 5.3, we have one of the following cases that we examine in sequence.

1. $\mathcal{Z}_C$ is non-crossing and non-overlapping. If $G_c = G$ then a call to PVMF-CENTER will return $\mathcal{Z}_C$. Otherwise $G_c \neq G$. Suppose $|\mathcal{Z}_C| = k1 < k = |\mathcal{Z}|$, and $f(\mathcal{Z}_C) = B' \leq B$. Among all non-crossing and non-overlapping partitions of G_c that have a value of the objective function smaller than or equal to B', $\mathcal{Z}_C$ must have the smallest cardinality otherwise $\mathcal{Z}$ is not optimal. Therefore to obtain all possible values for F_C and K_C, in this case, we call the procedure PVMF-CENTER(G_C, B', $k1$, F), for all possible values of B', $1 \leq B' \leq B$. Therefore we have at most B entries in each of F_C and K_C, and to calculate each entry we need $\mathcal{O}(n^8)$ processors and $\mathcal{O}(\log^2 n)$ time (theorem 6.2).

2. $\mathcal{Z}_C$ consists of a matching and cardinality one sets. In this case $G_c = G$ (see theorem 5.3), and both (x_1, y_1) and (x_p, y_q) are not in E. $\mathcal{Z}_C$ does not necessarily contain a maximum matching. By lemma 6.2 there are $\mathcal{O}(m)$ possibilities for the cardinality and the value of the objective function, for such an optimal partition $\mathcal{Z}_C$, that may be obtained in parallel in $\mathcal{O}(\log^2 n)$ time using $\mathcal{O}(mn)$ processors. Let these values be stored in F_C starting at $F_C(B+1)$ and the corresponding cardinality of the optimal partition be stored in K_C starting at $K_C(B+1)$.

3. $G_c = G$, both $(x_1, y_1$ and (x_p, y_q) are not in E, and $\mathcal{Z}_C$ has diameter three, four, or five and contains an Estar partitioned into two diameter two stars that form pattern 3.

The diameter of $\mathcal{Z}_C$ may be known in $\mathcal{O}(\log n)$ time and using $\mathcal{O}(n^2)$ processors (see lemma 4.10). For this case, we first prove the following claim. Suppose (Z_a, Z_b) overlap in pattern 3, Z_a is an X-star and Z_b is a Y-star (see lemma 5.10). Therefore (x_a^*, y_b^*) is a center of $Z_{ab} = Z_a \bigcup Z_b$, where x_a^* is the center of Z_a, and y_b^* is that of Z_b.

Claim 6.3.1 *Suppose (Z_a, Z_b) forms pattern 3, Z_a is an X-star, and Z_b is a Y-star. Suppose (s_b^X, s_b^Y) is not in E. If y_1 is in a cardinality one set then $s_a^Y = prev(x_a^*)$, and if x_1 is in a cardinality one set then $s_a^X = prev(y_b^*)$. Similarly if (t_b^X, t_a^Y) is not in E and y_q is in a cardinality one set then $t_a^Y = next(x_a^*)$, and if x_p is in a cardinality one set then $t_a^X = next(y_b^*)$,*

Proof of claim: If y_1 is in a cardinality one set, and $s_a^Y > prev(x_a^*)$, then $prev(x_a^*)$ is in a cardinality one set (refer to lemma 5.12, theorem 5.3, and lemma 5.13). But now $Z_a = Z_a \bigcup \{prev(x_a^*)\}$ has diameter two and there is a smaller partition.

The other cases follow similarly. ■

By lemma 5.12 each of x_1, y_1, x_p, and y_q is either in $Z_a \bigcup Z_b$ or is in a cardinality one set and has a neighbor in R_{ab}. Therefore we follow the following steps to find a minimum cardinality partition for G that satisfies condition 1.c in theorem 5.3.

(1) For every edge (x_a, y_b), in G, such that $(prev(x_a), prev(y_b))$ and $(next(x_a), next(y_b))$ are not in E, define sets $X_L(a,b) = \{x : x_1 \leq x < prev(y_b)\}$, and $X_R(a,b) = \{x : next(y_b) < x \leq x_p\}$. Define $Y_L(a,b)$ and $Y_R(a,b)$ similarly. Find an edge (x_{a1}, y_{b1}) such that $|X_L|+|X_R|+|Y_L|+|Y_R|$, where $X_L = X_L(a1, b1), X_R = \ldots$, is as small as possible. This can be accomplished in $\mathcal{O}(\log m)$ time using $\mathcal{O}(m^2)$ processors (m processors one for each edge and another m processors and $\mathcal{O}(\log m)$ time for the maximum [36]).

(2) Let $\alpha = 2^{\lceil \log(2+|X_L|+|X_R|+|Y_L|+|Y_R|) \rceil}$. Let $FM(1) = FM(2) = 2$, $FM(i) = 0$, $2 < i \leq 2 + |X_L| + |X_R| + |Y_L| + |Y_R|$, and all other values of $FM(i)$, $i \leq \alpha$, are set equal to the identity Δ. Let j be the index of the first empty entry in F_C. Let $K_C(j) = 2+|X_L|+|Y_L|+|X_R|+|Y_R|$. Then we calculate $F_C(j)$ by calling the procedure OBJ-FUN$(\alpha, FM, F_C(j))$.

Considering all cases above, it is clear there at most $m+B+1$ values for each of F_C and K_C. The cost of computing $\mathcal{Z}_C$ is that of calling algorithm 6.1 B times. ■

6.3 Computing the Left and Right Subpartitions, $\mathcal{Z}_L$ and $\mathcal{Z}_R$

Now we show how to calculate the value of the objective function and the cardinality for a partition, $\mathcal{Z}_L$, that satisfies one of the conditions 2.a, 2.b, 2.c, or 2.d in theorem 5.3. Let vertices x_L, y_L, and set S, where $|S^X| \leq 1$ and $|S^Y| \leq 1$, be such that $\mathcal{Z}_L$ is the partition of the subgraph, G_L, induced by the set $V_L = \{x : x \leq x_L\} \bigcup \{y : y \leq y_L\} \bigcup S$. Vertices in S are those vertices of $\mathcal{Z}_L$ that are in the range of $\mathcal{Z}_C$ (i.e. centers of diameter two stars that are trailers with respect to some set in $\mathcal{Z}_C$, see theorem 5.3). Recall that if both S^X and S^Y are non-empty then $\mathcal{Z}_L$ consists of two crossing diameter two stars, such that each is the trailer with respect to a set in $\mathcal{Z}_C$.

Let $G = (X, Y, E)$ be a biconvex graph. Let $\mathcal{Z}$ be an optimal partition for G that satisfies theorem 5.3. Let $\mathcal{Z}_L$ be a minimum cardinality partition for G_L, that satisfies one of the conditions 2.a, 2.b, 2.c, or 2.d in theorem 5.3.

Theorem 6.4 *There are $\mathcal{O}(m)$ possibilities for the cardinality and the value of the objective function for an optimal partition $\mathcal{Z}_L$ that may be computed in $\mathcal{O}(\log^2 n)$ time using $\mathcal{O}(m^2)$ processors.*

Proof : Let $S^X = \{s_X\}$, and $S^Y = \{s_Y\}$. Let x_L^* and y_L^* be the largest index vertices in V_L. Let F_L be an array that holds values of the function for all possible partitions, $\mathcal{Z}_L$, of G_L, and K_L holds the corresponding value of the cardinality of this partition.

Now referring to theorem 5.3, we have one of the following cases for $\mathcal{Z}_L$ that we examine in sequence.

1. $\mathcal{Z}_L$ consists of a matching (not necessarily a maximum one) and cardinality one sets. Using lemma 6.2, we can calculate all possible values for F_L and K_L in this case. There are $\mathcal{O}(m)$ such values that may be calculated using $\mathcal{O}(mn)$ processors and in $\mathcal{O}(\log^2 n)$ time.

2. G_L has diameter three, (x_1, y_1) is not in E, (x_L^*, y_L^*) is in E, and $\mathcal{Z}_L$ consists of two diameter two stars that induce an Estar. Furthermore if $S^X \neq \phi$ then it is adjacent to all vertices in G_L^Y, and if $S^Y \neq \phi$ then it is adjacent to all vertices in G_L^X (see theorem 5.3). This may be recognized in $\mathcal{O}(\log n)$ time using $\mathcal{O}(n)$ processors. Let $F_L(m+1) = 2 \bullet 2$ and $K_L(m+1) = 2$.

3. G_L has diameter three or four, (x_1, y_1) is not in E, $S = \phi$, $(x_L, y_L) = (x_L^*, y_L^*) \in E$, $\mathcal{Z}_L$ contains two sets that overlap in pattern 3 (see theorem 5.3).

   Suppose (Z_a, Z_b) are sets in $\mathcal{Z}_L$ that form pattern 3.

   By theorem 5.3 each of x_1 and y_1 is either in $Z_a \bigcup Z_b$ or is in a cardinality one set and has a neighbor in R_{ab}. Therefore we follow the following steps to find a minimum cardinality such partition for G_L in this case. We will use the result of claim 6.3.1.

   (1) For every edge (x_a, y_b), in G_L, such that $next(x_a) = y_L$, $next(y_a) = x_L$, and $(prev(x_a), prev(y_b))$ is not in E, define the set $X_L(a, b) = \{x : x_1 \leq x < prev(y_b)\}$, and define $Y_L(a, b)$ similarly. Among all such edges, find an edge (x_{a1}, y_{b1}) such that

$|X_L| + |Y_L|$, where $X_L = X_L(a1, b1)$ and $Y_L = Y_L(a1, b1)$, is as small as possible. This can be accomplished in $\mathcal{O}(\log m)$ time using $\mathcal{O}(m^2)$ processors (m processors one for each edge and another m processors and $\mathcal{O}(\log m)$ time for the maximum [36]).

(2) Let $\alpha = 2^{\lceil \log(2+|X_L|+|Y_L|) \rceil}$. Let $FM(1) = FM(2) = 2$, $FM(i) = 0$, $2 < i \leq 2 + |X_L| + |Y_L|$, and all other values of $FM(i)$, where $i \leq \alpha$ are set equal to the identity Δ. Let j be the index of the first empty entry in F_L. Let $K_L(j) = 2 + |X_L| + |Y_L|$. Then we calculate $F_L(j)$ by calling the procedure OBJ-FUN($\alpha, FM, F_L(j)$).

4. $\mathcal{Z}_L$ consists of a diameter two star (case (d) in theorem 5.3), (x_1, y_1) is not in E, $|S| = 1$, and S is the center of the star.

   In this case $F_L(i) = 2$, and $K_L(i) = 1$, where i is the first empty entry in F_L.

Therefore, in total, there are at most $m + 1$ values for each of F_L and K_L (since no two of the last three cases can simultaneously occur). Recognizing each case merely involves calculating the diameter of the subgraph induced by $\mathcal{Z}_L$, and therefore requires no more than $\mathcal{O}(n^2)$ processors and $\mathcal{O}(\log n)$ time. The complexity of obtaining all possible values is that of case 3. ■

Let $G = (X, Y, E)$ be a biconvex graph. Let $\mathcal{Z}$ be an optimal partition for G that satisfies theorem 5.3. Let $\mathcal{Z}_R$ be a smallest cardinality subpartition of $\mathcal{Z}$ that satisfies one of the conditions 3.a, 3.b, 3.c, or 3.d in the statement of that theorem. Suppose we are given G_R the subgraph of G induced by sets in $\mathcal{Z}_R$, and G_R is non-empty. Then we have a theorem analogous to theorem 6.4.

Theorem 6.5 *There are $\mathcal{O}(m)$ possibilities for the cardinality and the value of the objective function for an optimal partition $\mathcal{Z}_R$ that may be computed in $\mathcal{O}(\log^2 n)$ time using $\mathcal{O}(m^2)$ processors.*

6.4 Parallel Algorithm for $PVMF$

Theorem 6.6 *Given a biconvex graph G, and integer B, an optimal partition for G may be calculated in $\mathcal{O}(\log^2 n)$ time using $\mathcal{O}(Bn^{16})$ processors.*

Proof : We perform the following procedure. In this procedure we will try all possibilities for the subgraphs G_L, G_R, and G_C. G_L will be the subgraph induced by vertices in sets in $\mathcal{Z}_L$, G_R will be the subgraph induced by vertices in sets in $\mathcal{Z}_R$, and G_C will be the subgraph induced by vertices in sets in $\mathcal{Z}_C$.

begin

for all x_L, y_L, x_R, y_R, u_L, w_L, u_R, w_R, *where* $x_L < u_L < u_R < x_R$, $y_L < w_L < w_R < y_R$, *(possibly one or more of these vertices does not exist)* **do in parallel**

$G_L = [\{x, y : x < x_L, y < y_L\} \bigcup \{u_L, w_L\}]$

$G_R = [\{x, y : x > x_R, y > y_R\} \bigcup \{u_R, w_R\}]$

/Vertices u_L, w_L, u_R, and w_R are centers of trailers in $\mathcal{Z}_L$ or $\mathcal{Z}_R$, that overlap some set in $\mathcal{Z}_C$/

$G_C = G - (G_L \bigcup G_R)$

Find all possible F_L and the corresponding K_L for G_L, as in theorem 6.4

Find all possible F_R and the corresponding K_R for G_R, as in theorem 6.5

Find all possible F_C and the corresponding K_C for G_C, as in theorem 6.3

$k = \min_{i,j,l}(K_L(i) + K_C(j) + K_R(l))$, *where* $F_L(i) \bullet F_C(j) \bullet F_R(l) \leq B$

endfor

end

■

6.5 Sequential Algorithm for PVD on Bipartite Permutation Graphs

We now present a sequential algorithm for PVD (Partitioning vertices to minimize maximum diameter) on bipartite permutation graphs, of a lower complexity than the sequential version of the parallel algorithm. Recall that bipartite permutation graphs have an optimal non-crossing non-overlapping partition (corollary 5.3).

Partitioning the vertex set of a graph into the minimum number of cliques (i.e. PVC), a special case of PVD, has a linear time sequential algorithm and an NC parallel algorithm for bipartite permutation graphs (see [38] and [20] respectively).

We prove a useful property for optimal non-overlapping non-crossing partitions, when the input graph is a bipartite permutation graph. With very few modifications such a partition may be transformed into one with sets having nearly equal diameters. We use the above result and *Depth First Search (DFS)* to design the desired algorithm. The algorithm is similar to algorithm 3.2.

Let $G = (X, Y, E)$ be a bipartite permutation graph. Let vertices of $X \bigcup Y$ be strongly ordered (see Appendix B). The next definitions and propositions are essential for the algorithm.

For any vertex v, let $next^0(v) = v$, and for $l > 0$ let $next^l(v) = next(next^{l-1}(v))$. Let $e_u = (u^X, u^Y)$ be an edge in E, $e_u^X = u^X$ and $e_u^Y = u^Y$. Define $next(e_u) = (next(u^X), next(u^Y))$.

Lemma 6.3 *For any pair of vertices u and v, $dist(u, v) = B$ if and only if u is adjacent to $next^{B-1}(v)$ or v is adjacent to $next^{B-1}(u)$, where B is the smallest such integer.*

Proof: By the connectivity of the graph, v is adjacent to $next^{b-1}(u)$ for some b. Let b have the smallest index possible. Therefore $dist(u,v) \leq b$. We now show that $dist(u,v) = b$.

Consider the path $P_1 = \{u, next(u), next^2(u), \ldots, next^{b-1}(u), v\}$, where b is the smallest such integer.

If $b \leq 3$ or $v = next^b(v)$ then it is clear that P_1 is a shortest path between u and v.

Now suppose $b > 3$. It is clear that v has no neighbor smaller than or equal to $next^{b-3}(u)$, by our choice of b. Also no neighbor, w, of v has a neighbor smaller than or equal to $next^{b-4}(u)$, otherwise w equals $next^{b-3}(u)$ or v is adjacent to $next^{b-3}(u)$ (respectively, if w is larger than or equal to $next^{b-3}(u)$ or if w is smaller than $next^{b-3}(u)$). Therefore P_1 is a shortest path between u and v. Since P_1 has length b the result now follows. ■

> **Definition 6.1:** A set Z_i is said to be *maximal*, if $Z_i^Y \neq \phi$ and $Z_i^X \neq \phi$ and there are no vertices x^* and y^* such that $y^* > t_i^Y$ and $x^* \geq t_i^X$, or $y^* \geq t_i^Y$ and $x^* > t_i^X$, and $diam(Z_i \bigcup \{y : t_i^Y < y \leq y^*\} \bigcup \{x : t_i^X < x \leq x^*\}) \leq diam(Z_i)$.

Let Z_i be a connected set of vertices, that is not necessarily member in an optimal partition.

Lemma 6.4 *If $diam(Z_i) > 0$ and $(t_i^X, t_i^Y) = next^{B-1}(s_i^X, s_i^Y)$, then $diam(Z_i) = B$ and Z_i is maximal.*

Proof: By strong ordering of the vertices and the fact that Z_i is connected, both (s_i^X, s_i^Y) and (t_i^X, t_i^Y) are edges in G.

Let $e_s = (s_i^X, s_i^Y)$ and $e_t = (t_i^X, t_i^Y)$. We prove the claim by induction on B.

The claim is true if $B = 1$.

Suppose it is true for all $B < b$. Let Z_i be such that $(t_i^X, t_i^Y) = next^{b-1}(s_i^X, s_i^Y)$. Let $Z_i' = Z_i - \{v : v > next^{b-2}(s_i^X, s_i^Y)\}$. By the induction hypothesis Z_i' is a maximal set of diameter $b-1$. Since (t_i^Y, t_i^X) is in E, each of t_i^Y and t_i^X is either adjacent to an endpoint of $next^{b-2}(s_i^X, s_i^Y)$ or is an endpoint of $next^{b-2}(s_i^X, s_i^Y)$, therefore $b - 1 < diam(Z_i) \leq b$.

Suppose Z_i is not maximal. Then there are vertices x' and y', such that $diam(Z_i \bigcup U) \leq b$ where $U = \{x, y : t_i^X < x \leq x', t_i^Y < y \leq y'\}$, and either $x' \geq t_i^X$ and $y' > t_i^Y$, or $x' > t_i^X$ and $y' \geq t_i^Y$ (by definition 6.1).

There is at least one vertex in U that has a neighbor in Z_i. Suppose that vertex is x_a, let x_a have the largest index possible. By strong ordering of the vertices, x_a is adjacent to t_i^Y. x_a is not adjacent to $next^{b-2}(s_i^Y)$ or $next^{b-2}(s_i^X)$, otherwise $x_a = t_i^X$. t_i^Y equals $next^{b-1}(s_i^Y)$ or $next^{b-1}(s_i^X)$, but not both. Therefore x_a is within distance b of at most one of s_i^Y or s_i^X but not both by lemma 6.3, and it cannot be true that $diam(Z_i \bigcup U) \leq b$.

Therefore $diam(Z_i) = b$ and Z_i is maximal. ■

Proposition 6.1 *Given edge $e_s = (s_i^X, s_i^Y)$, the maximal set Z_i that starts at e_s and has diameter D is* unique*.*

Proposition 6.2 *Let Z_r be a subset of some set Z_i, such that the first vertices in Z_r are adjacent, the last vertices in Z_r are adjacent, and Z_r consists of consecutive vertices in Z_i, then Z_r has a diameter not larger than that of Z_i.*

Proof: By lemma 4.7 and the fact that all shortest paths in Z_i are S-paths. ∎

Given an optimal non-crossing non-overlapping partition $\mathcal{Z}$ for G.

Theorem 6.7 (Near Equal Diameters Partition) *G has a unique optimal non-crossing non-overlapping partition, where each set is maximal and satisfies the following.*
(a) Each set, except possibly for the last (that which contains x_p, y_q or both), has diameter B or $B-1$.
(b) If $B=1$ then the endpoints of the first edge in G are in the same set.
(c) If $B>1$ and there is a set Z_a that has diameter $B-1$ then exactly one of the following is true. s_{a+1}^X is adjacent to t_a^Y and $Z_{a+1} - \{s_{a+1}^X\}$ is disconnected, or s_{a+1}^Y is adjacent to t_a^X and $Z_{a+1} - \{s_{a+1}^Y\}$ is disconnected.

Proof : Let $\mathcal{Z}$ be an optimal partition for G that satisfies corollary 5.3 (i.e. is non-crossing and non-overlapping). We will consider the cases $B=1$ and $B>1$. Note that in any (connected) set of vertices, of diameter greater than zero, the smallest index vertices are adjacent, and the largest index vertices are adjacent (by strong ordering).

Case 1: $B=1$.

It is clear that any set has diameter B or $B-1$. We prove the claim by induction on the number of the vertices in the graph. We will have to consider possibly disconnected graphs. The claim is true for graphs having two or fewer vertices; the optimal partition for such a graph is unique. Suppose it is true for all graphs having fewer than n vertices.

Let G be a graph that has n vertices. Let (x_{i1}, y_{j1}) be the first edge in G. If $x_{i1} \neq x_1$ or $y_{j1} \neq y_1$ then the graph is not connected, and no vertex smaller than x_{i1} or y_{j1} has neighbors in G (otherwise, by strong ordering, there is a contradiction to our choice of (x_{i1}, y_{j1})). Therefore all such vertices are in cardinality one sets in $\mathcal{Z}$.

By the induction hypothesis the claim holds for the subgraph of G induced by vertices greater than or equal to x_{i1} or y_{j1}. Therefore it suffices to consider the case when $x_{i1} = x_1$ and $y_{j1} = y_1$.

Suppose x_1 and y_1 are not in the same set. They cannot both be in diameter one sets otherwise $\mathcal{Z}$ is crossing. They cannot both be in diameter zero sets otherwise there is a smaller partition.

Therefore suppose x_1 is in a diameter one set $Z_i = \{x_1, y_i\}$, and y_1 is in a diameter zero set Z_j. All vertices smaller than y_i are in cardinality one sets, and $y_i > y_1$. Let $Z_i = \{x_1, y_1\}$, and $Z_j = \{y_i\}$.

Let G' be the subgraph induced by vertices larger than x_1 or y_i. By the induction hypothesis G' has an optimal partition that satisfies the theorem. Therefore that subpartition of G' together with Z_i and all cardinality one sets smaller than or equal to y_i is an optimal partition for G that satisfies the theorem.

The first set in any such partition is unique (see proposition 6.1). Therefore, by the induction hypothesis, the whole partition is unique.

Case 2: $B > 1$.

It is clear that the hypothesis is true if $|\mathcal{Z}| = 1$. Suppose it is true for all optimal partitions of cardinality smaller than k. Let $\mathcal{Z}$ be a cardinality k partition that has the largest number $a-1$ of consecutive sets $Z_1, Z_2, \ldots, Z_{a-1}$ that satisfy the theorem. Therefore $diam(Z_a) < B$, and Z_a is not the last set. If $a > 1$, then by the induction hypothesis, the graph induced by all sets greater than $a-1$ has an optimal partition that satisfies the theorem. Then it must be true that $a = 1$. Let $Z_a' = \{x, y : x, y \leq next^{B-1}(x_1, y_1)\}$. Z_a' is maximal by proposition 6.4. Let $Z_i' = Z_i - Z_a'$ for all Z_i in $\mathcal{Z}$. If $diam(Z_i') \leq diam(Z_i)$ for all i, then we have an optimal partition that agrees more with the theorem, a contradiction. Otherwise let x' and y' be the vertices that immediately follow the endpoints of $next^{B-1}(x_1, y_1)$.

If both x' and y' are in some set Z_b, then it must be true that they are not adjacent. At least one of x' or y' has a neighbor in Z_b', by strong ordering. Suppose x' has a neighbor in Z_b', while y' does not. Then y' is adjacent to, $t_a'^X$, the last vertex in Z_a', and $t_a'^X$ is in $Z_a' \bigcap Z_b$. Therefore $diam(Z_a' - \{t_a'^X\}) \leq diam(Z_a')$ (otherwise Z_a' is not maximal). Let $Z_a' = Z_a' - \{t_a'^X\}$, and $Z_b' = Z_b' \bigcup \{t_a'^X\}$. Consequently the optimal partition $\{Z_1, \ldots, Z_{a-1}, Z_a', Z_b', \ldots\}$ agrees more with the theorem, a contradiction.

Otherwise let x' be in Z_b and y' be in Z_c, where $c \neq b$, and suppose $s_c^X > x'$ and $s_b^Y < y'$ (since $\mathcal{Z}$ is non-overlapping and non-crossing). If $diam(Z_c') > diam(Z_c)$ then y' is not adjacent to s_c^X. s_c^X has a neighbor in Z_a' and by strong ordering is adjacent to $t_a'^Y$, the last vertex in $Z_a'^Y$. Since $\mathcal{Z}$ is non-overlapping and non-crossing, therefore $t_a'^Y$ is in Z_c. By convexity, $t_a'^Y$ is adjacent to all vertices x, $x' \leq x \leq s_c^X$. It must be true that $diam(Z_a' - \{t_a'^Y\}) \leq diam(Z_a')$, otherwise by lemma 4.6 we can add all neighbors of $t_a'^Y$ to Z_a' without increasing its diameter, implying that Z_a' is not maximal. Let $Z_a' = Z_a' - \{t_a'^Y\}$, and $Z_c' = (Z_c \bigcup Z_b) - Z_a'$. Note that the first vertices in Z_c' are $t_a'^Y$ and x' and they are adjacent. The last vertices in Z_c' are t_c^Y and t_c^X. Since $dist(x', v) \leq dist(s_c^X, v)$ for all v in Z_c', then $diam(Z_c') \leq diam(Z_c)$ (also by proposition 6.2). But now $\mathcal{Z} - \{Z_a, Z_b, Z_c\} \bigcup \{Z_a', Z_c'\}$ is an optimal partition of smaller cardinality, a contradiction. Otherwise $diam(Z_b') > diam(Z_b)$. Therefore $t_b^X > x'$. Let $Z_a' = Z_a' - \{t_a'^Y\}$ and $Z_b' = \{t_a'^Y\} \bigcup \{x : x' \leq x \leq t_b^X\}$. As above it must be true that Z'_a has diameter $B-1$. Z'_b has diameter two. Now we have an optimal partition that agrees more with the theorem, a contradiction.

Note that if $diam(Z_i) = B-1$, for some i, then the following cannot be true: $Z_{i+1} - \{s^X_{i+1}\}$ is disconnected or $Z_{i+1} - \{s^Y_{i+1}\}$ is disconnected, and both s^X_{i+1} and s^Y_{i+1} have neighbors in Z_i. Otherwise by strong ordering all neighbors of s^X_{i+1} are adjacent to all neighbors of s^Y_{i+1}, and vice versa; thus $diam(Z_{i+1} - \{s^X_{i+1}\}) \leq diam(Z_{i+1})$ and $diam(Z_{i+1} - \{s^Y_{i+1}\}) \leq diam(Z_{i+1})$.

Therefore there is an optimal partition that has a larger number of consecutive sets that satisfy the theorem, which contradicts our choice of $\mathcal{Z}$.

Since Z_1 is unique by proposition 6.1, then $\mathcal{Z}$ is unique. ■

We now present a sequential algorithm that uses the result of the above theorem and DFS to find a non-overlapping partition. We give an overview of the algorithm.

The algorithm we present greedily finds maximal sets of nearly equal diameter as in theorem 6.7. Let G_1 be the subgraph of G not partitioned so far. Let e_0 be the first edge in G_1, i.e. the edge whose endpoints have the smallest indices.

The algorithm starts at e_0 and finds a set that extends as far as possible from e_0 (maximal sets of diameter B), following the edge sequence e_0, $next(e_0)$, $next(next(e_0)), \ldots$. To find a set that extends the farthest, DFS is used, where edges in the DFS sequence are e_0, $next(e_0)$, $next(next(e_0)), \ldots next^{B-1}(e_0)$ (see lemma 6.4). On returning from the DFS subroutine, the remaining subgraph is examined to determine whether or not it is connected. If it is connected then the smallest edge of that subgraph is stored in the array T, that edge marks the beginning of the next set. If the remaining subgraph is disconnected and $B > 1$, then one vertex is taken from previous set to connect it. That vertex is one of the end vertices of the remaining subgraph (refer to theorem 6.7). If it is disconnected and $B = 1$ then all cardinality one components are added to the partition as cardinality one sets. Therefore at the end we have the partition: $\mathcal{Z} = \{Z_1, Z_2, \ldots, Z_k\}$, where Z_i is the set that starts at the edge $T(i)$ and contains all vertices between the endpoints of $T(i)$ and up to, but not including, the endpoints of $T(i+1)$. If Z_i is a cardinality one set, then $T(i)$ holds Z_i.

Let Λ be the *null vertex*, and (Λ, Λ) be the *null edge*. We consider vertices to be edges of the form $e = (\Lambda, y)$ or $e = (x, \Lambda)$. We assume that vertices of the bipartite permutation graph $G = (X, Y, E)$ are strongly ordered.

Algorithm 6.2 Algorithm PVD-BPG(G, B, k, V)

Input: *Integer B and $G = (X, Y, E)$ where G is a strongly ordered BPG .*
Output: *The cardinality k of the optimal non-overlapping non-crossing partition, $k = |\mathcal{Z}|$.*

begin

1. **for all** $v \in V$ **do**
   unmark v
   endfor

2. $T(index) = \Lambda$, $1 \leq index \leq n$

3. $e_0 = (x_1, y_1)$

4. $G_1 = G$.

5. $index = 1$

6. **while** $e_0 \neq (\Lambda, \Lambda)$ **do**
   Add to T $\{u : u \in G_1, u < e_0^X$ *or* $u < e_0^Y\}$, *each vertex in a separate entry, update index*
   $T(index) = e_0$
   $index = index + 1$
   Call DFS(G, B, e_0)
   $G_1 = \{v : v \geq e_0^X$ *or* $v \geq e_0^Y\}$
   if G_1 *is disconnected and* $B > 1$ **then**
   Let $e_0 = (e_0^X - 1, e_0^Y)$ or $e_0 = (e_0^X, e_0^Y - 1)$
   /Note that exactly one of the above choices qualifies for e_0 by theorem 6.7/
   endif
   enddo

7. $k = index - 1$
   end

Procedure DFS(G, B, e_0)
begin

1. $e = e_0$

2. $curr = (\Lambda, \Lambda)$

3. $it = 1$

4. **while** $e \neq (\Lambda, \Lambda)$ *and* $it \leq B - 1$ **do**
   mark e
   $curr = e$
   $e = next(e)$
   $it = it + 1$
   enddo
   /Note that $curr = next^{B-1}(e_0)$/

5. $e_0 = (curr^X + 1, curr^Y + 1)$
   end

Theorem 6.8 *Algorithm 6.2 computes the cardinality, k, of the optimal partition that satisfies theorem 6.7 in $\mathcal{O}(n)$ time.*

Proof : To prove the correctness of the algorithm, it suffices to prove that the partition that the algorithm outputs satisfies theorem 6.7.

We prove the correctness of the algorithm by induction on k.

By lemma 6.4, the algorithm is correct for all graphs that have an optimal partition of cardinality one.

Suppose the algorithm is correct for all graphs that have an optimal partition of cardinality smaller than k, that satisfies theorem 6.7. Let $\mathcal{Z}^*$ be the partition of G that satisfies theorem 6.7, and let $\mathcal{Z}$ be the partition returned by the algorithm. Let a be the largest number of consecutive sets $Z_1^*, Z_2^*, \ldots, Z_a^*$, in $\mathcal{Z}^*$, that agree with sets in $\mathcal{Z}$, i.e. $Z_i = Z_i^*$ for all $i \leq a$, and $Z_{a+1}^* \neq Z_a$.

Therefore the subgraph induced by the sets Z_j^*, $j > a$ has a $k - a$ optimal partition that satisfies theorem 6.7. If $a > 0$ then by the induction hypothesis the algorithm performs correctly.

Therefore suppose $a = 0$. If $diam(Z_1) = B$, then by proposition 6.1 it must be true that $Z_1 = Z_1^*$. If $diam(Z_1) < B$ then it must be true that $diam(Z_1) = B - 1$, and $Z_2 - \{s_2^X\}$ is disconnected or $Z_2 - \{s_2^Y\}$ is disconnected. Suppose $Z_2 - \{s_2^X\}$ is disconnected and s_2^X has a neighbor is Z_1. Therefore $Z_1 \bigcup \{s_2^X\}$ is a maximal set of diameter B.

$Z_1 \neq Z_1^*$ implies that $Z_1^* = Z_1 \bigcup \{s_2^X\}$. But now Z_2^* is disconnected or has cardinality one, both contradict our choice of $\mathcal{Z}^*$. Therefore it must be true that $Z_1 = Z_1^*$.

This completes the correctness proof. We now prove the complexity.

To test that a subgraph is connected it suffices to test (in linear time) whether or not its end vertices have some neighbors in that subgraph.

We now calculate the complexity of DFS. Let e_0^i be the value of e_0 upon entry to DFS for the ith time. Every vertex v between any of the end vertices of e_0^i and e_0^{i+1} (exclusive), is scanned at most once by DFS. The only vertices scanned more than once are those endpoints of e_0^i, for $i \geq 2$. Since each such endpoint marks the beginning of a new set, therefore it is scanned at most twice. Summing over all calls to DFS, the complexity of the algorithm is $\mathcal{O}(n)$. ■

Chapter 7

Algorithms for Best Cluster

7.1 Introduction

In this chapter we study the second of the pair of clustering problems that this thesis deals with. In contrast with the first problem (or more precisely, set of problems) which seeks a partition of the vertex set, the second problem seeks one subgraph of the given graph. The target subgraph has a fixed number of vertices and induces the maximum possible number of edges. We called this problem CXE. We present an efficient parallel algorithm for the problem CXE for a class of graphs that intersects unit interval graphs. Recall the statement of the problem.

> **Problem CXE : Cluster with the Maximum Number of Edges**
>
> Given a graph $G = (V, E)$ and integers B and k, does G contain a connected induced subgraph on k vertices (a k-cluster) that has at least B edges?

We first review definitions of the graph families we deal with in this chapter.

An *interval graph*, IG, is the intersection graph of intervals in the real line. If the intervals have equal lengths then the intersection graph is a *unit interval graph* (UIG). $UIGs$ are also characterized as those IGs that contain no induced subgraph isomorphic to $K_{1,3}$, which is the complete bipartite graph, $G = (X, Y, E)$, with $|X| = 1$ and $|Y| = 3$. Figure 7.1 shows examples of IGs and $UIGs$, together with their respective interval representations. It is easy to see that induced subgraphs of IGs (or of $UIGs$) are IGs (respectively $UIGs$). As mentioned in the introductory chapter, interval and unit interval graphs arise in many applications. For such application examples see [68] and [40].

In this chapter we will refer to interval i, interval $[a_i, b_i]$, and vertex v_i or i, interchangeably.

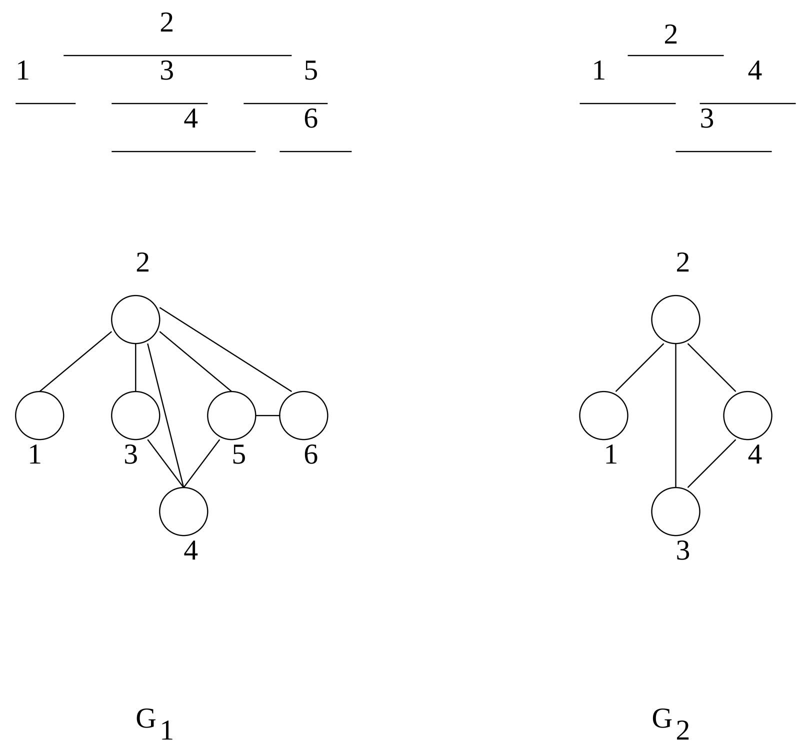

Figure 7.1: An interval graph G_1 and a unit interval graph G_2

For any interval graph $G = (V, E)$, the set of all maximal cliques of G, $\mathcal{C}$, has cardinality $\mathcal{O}(n)$ [34]. The *clique matrix*, A, of G is one whose rows represent maximal cliques and columns represent vertices; entry $A(i, j)$ is set to one if vertex j belongs to clique i. The clique matrix of an interval graph has the property that its rows and columns can be arranged so that it has the *consecutive ones* property for its columns; i.e. cliques to which any vertex belongs occur consecutively, see [32] and [73]. When G is a unit interval graph then A has the consecutive ones property for both rows and columns; i.e. cliques to which any vertex belongs are consecutive and cliques consist of consecutive vertices. Figure 7.2 shows the clique matrices for the interval graphs in figure 7.1.

$$
\begin{array}{c}
\text{vertices} \rightarrow \\
\begin{array}{c} C_1 \\ C_2 \\ C_3 \end{array} \\
\uparrow \\
\text{cliques}
\end{array}
\begin{array}{c}
\begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\
\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 \end{bmatrix} \\
A
\end{array}
\qquad
\begin{array}{c}
\\
\begin{array}{c} C_1 \\ C_2 \end{array}
\end{array}
\begin{array}{c}
\begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\
\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \\
B
\end{array}
$$

Figure 7.2: Clique matrix for G_1 (matrix A), and for G_2 (matrix B)

There are efficient parallel algorithms to obtain an interval representation for a graph (if one exists), such as that given in [56]. The set of intervals may be sorted efficiently in parallel [36]. We will, henceforth, assume that we are given a unit interval graph whose intervals are sorted so that $v_i = [a_i, a_i + 1] < v_j = [a_j, a_j + 1]$ if $a_i < a_j$, otherwise ties are broken arbitrarily.

This chapter is organized as follows. The second section introduces some preliminaries and definitions. In the following sections we characterize *Path Unit Interval graphs PUIGs* which intersect *UIGs* (section three), prove some useful properties of a best cluster on these graphs (section four), and present a parallel algorithm to solve CXE on these graphs (section five). Since the only polynomial time sequential algorithms known for this problem are for cographs [21] and for partial h-trees where h is fixed [54], this algorithm might contribute to our understanding of the problem.

7.2 Preliminaries and Definitions

We always deal with connected graphs. Given a UIG, $G = (V, E)$. Let $n = |V|$ and $m = |E|$.

Let the *neighborhood set* of vertex v be $N(v) = \{v' : (v, v') \in E\}$, let $N[v]$ be the *closed neighborhood set* of v, $N[v] = \{v\} \bigcup N(v)$.

Intervals are ordered so that $[a, a+1] < [b, b+1]$ if $a < b$. Ties are broken arbitrarily. Let vertices of G be ordered accordingly. Note that vertices of G satisfy: if v_i is adjacent to v_j, $i < j$, then v_i is adjacent to all v_l, $j \leq l < i$.

Let the set of the maximal cliques of G be $\mathcal{C}$, then members of $\mathcal{C}$ may be ordered so that: $C_1 = \{v_1, v_2, \ldots, v_{i_1}\}$, $C_2 = \{v_{i_2}, v_{i_3+1}, \ldots, v_{i_3}\}, \ldots, C_q = \{v_{i_{2(q-1)}}, \ldots.. v_n\}$, where $q = |\mathcal{C}|$, and $v_{i_j} \geq v_{i_{j+1}}$ for all odd values of j, $1 \leq j \leq 2(q-1)$ (since the graph is connected). It is easy to show that $q \leq n$. Define the following.

The *area set* $\mathcal{I}$ of G, $\mathcal{I} = \{I_1, I_2, \ldots I_p\}$ where v_i and $v_j \in I_l$ if and only if $N[v_i] = N[v_j]$. Note that $\mathcal{I}$ is a partition of V. Let $min(I) = \min_i v_i \in I$. Then members of $\mathcal{I}$ may be ordered so that $I_i < I_j$ if $min(I_i) < min(I_j)$. It is clear that $p \leq n$, each area induces a clique in G, and maximal cliques consist of consecutive areas.

The *capacity* of area I, $c(I)$, is equal to $|I|$.

Let $N(I) = \bigcup_{v \in I}(N(v) - I)$ and $N[I] = I \bigcup N(I)$.

I_i is said to be *dominated by* I_j if $N[I_i] \subset N[I_j]$, I_j is said to *dominate* I_i. I_j is said to be a *dominating* area if there is no area I_l such that I_j is dominated by I_l.

I_i and I_j are said to be *adjacent* if there are adjacent vertices v_i and v_j where $v_i \in I_i$ and $v_j \in I_j$.

Define the *prime graph* of G, $G' = (\mathcal{I}, \mathcal{E})$, such that $(I_i, I_j) \in \mathcal{E}$ if I_i is adjacent to I_j.

Figure 7.3 shows a UIG, a representation of its maximal cliques as sets in a *Venn diagram*, some of its areas, and its prime graph.

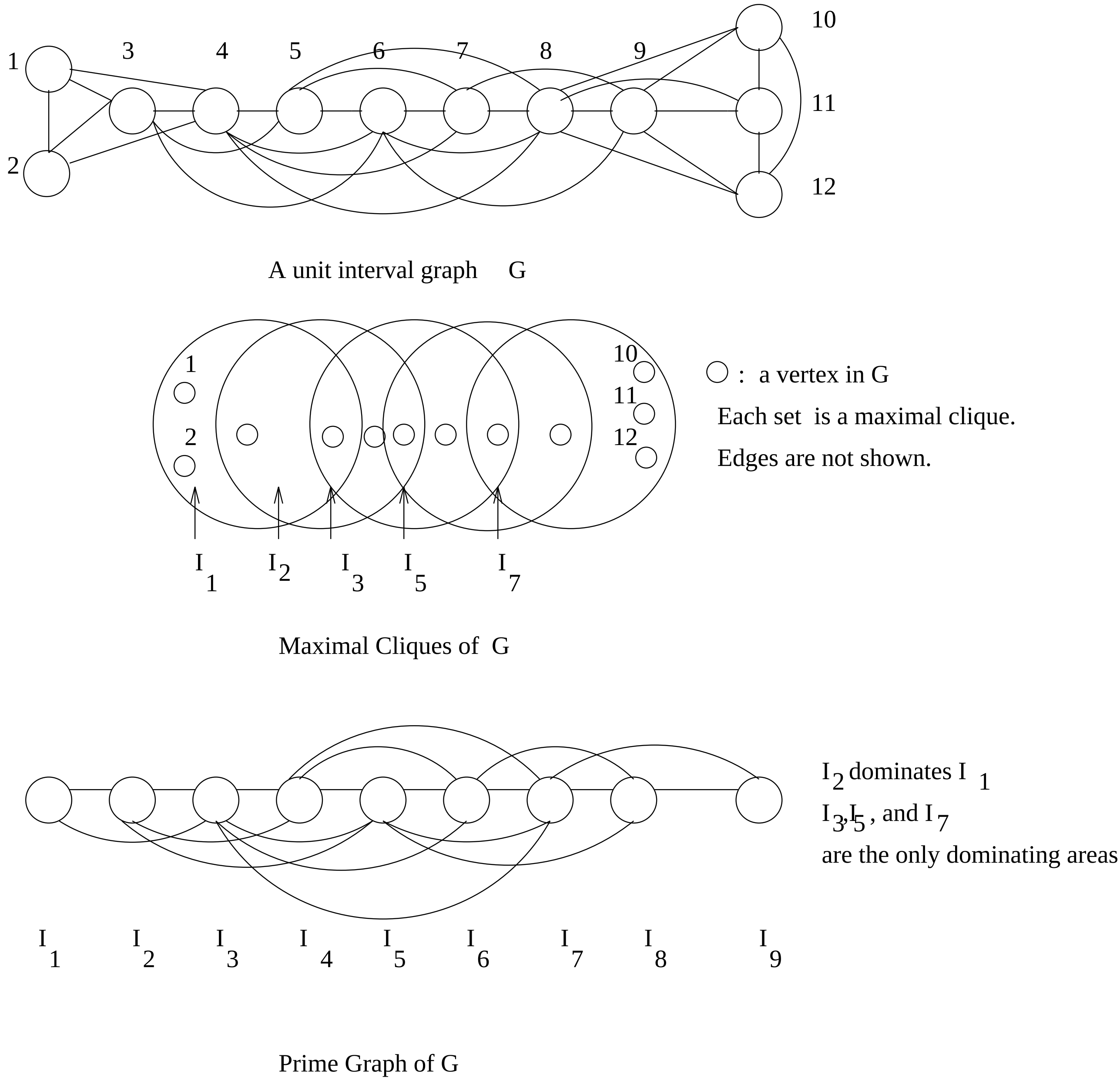

Figure 7.3: A unit interval graph, its maximal cliques and its prime graph

The following lemma shows that there is an efficient parallel algorithm to obtain the prime graph representation for $UIGs$.

Let v be any vertex in V, define $f(v) = \min_j v_j \in N[v]$ and $l(v) = \max_j v_j \in N[v]$. Therefore $N[v] = \{u : f(v) \leq u \leq l(v)\}$. For any vertex v, $f(v)$ and $l(v)$ can be computed in parallel in $\mathcal{O}(\log n)$ time and using $\mathcal{O}(n)$ processors using a well-known technique of finding the minimum or the maximum of a list of unsorted numbers [36]. To compute $l(v)$ and $f(v)$ for all vertices v, we need at most $\mathcal{O}(\log n)$ time and $\mathcal{O}(n^2)$ processors. Suppose we already computed $l(v)$ and $f(v)$ for all vertices v.

Lemma 7.1 (Prime Graph Representation) *Given a UIG $G = (V, E)$, its prime graph $G' = (\mathcal{I}, \mathcal{E})$ can be constructed in parallel in $\mathcal{O}(\log n)$ time with $\mathcal{O}(n^2)$ processors.*

Proof : We present the following parallel algorithm. Initially each area I_i is set to v_i (step 1 of the algorithm). Notice that vertices v_i and v_j are in the same area if and only if $f(v_i) = f(v_j)$ and $l(v_i) = l(v_j)$. In step 2 contents of I_i are changed. If a vertex v_i is the smallest vertex in an area that area is set to v_i, i.e. $I_i = \{v_i\}$ (step 2), $v_i = f(I_i)$, otherwise area I_i is emptied (step 3). Therefore non-empty areas are not consecutive but are still linearly ordered. Since areas are disjoint and consist of consecutive vertices, it suffices to store the smallest vertex in an area. The capacity of an area is calculated in steps 4 and 5. The output is areas: $I_{i1}, I_{i2}, \ldots$, and capacities: $c(I_{i1}), c(I_{i2}), \ldots$. Let $l(i) = l(I_i)$, then $I_{ij} = \{v_{ij}, v_{ij+1}, \ldots, v_{l(ij)}\}$ and $I_{i1} < I_{i2} < \cdots$.

Algorithm 7.1 Algorithm Prime-Graph

Input: $G = (V, E)$, $f(v)$ *and* $l(v)$ *for all vertices* v *in* V
Output: $G' = (\mathcal{I}, \mathcal{E})$, $c(I)$ for all $I \in \mathcal{I}$

1. **for all** i **do in parallel**
   $I_i = v_i$
   endfor
2. **for all** i **do in parallel**
   $I_i = \min_{j, l(v_j) = l(v_i), f(v_j) = f(v_i)} v_j$
   endfor
3. **for all** i **do in parallel**
   if $I_i \neq i$ **then** $I_i = \phi$
   endfor
4. **for all** i *s.t.* $I_i \neq \phi$ **do in parallel**
   $l(i) = \min_{j, j > i, I_j \neq \phi} I_j - 1$
   endfor
5. **for all** i *s.t.* $I_i \neq \phi$ **do in parallel**
   $c(I_i) = l(i) - i + 1$

endfor

Since each loop requires $\mathcal{O}(n)$ processors, one for each i, and each min operation requires $\mathcal{O}(n)$ processors and $\mathcal{O}(\log n)$ time, therefore the algorithm requires $\mathcal{O}(n^2)$ processors and $\mathcal{O}(\log n)$ time. ∎

7.3 Characterization of Path Unit Interval Graphs ($PUIGs$)

In this section we characterize the class of graphs for which we present an efficient parallel algorithm to solve CXE.

Definition 7.1: Given a UIG, $G = (V, E)$, G is a $PUIG$ if its prime graph, G', does not contain an induced subgraph isomorphic to the graph G'_1 shown in figure 7.4. Note that in this induced subgraph I_3 and I_4 are the only dominating areas.

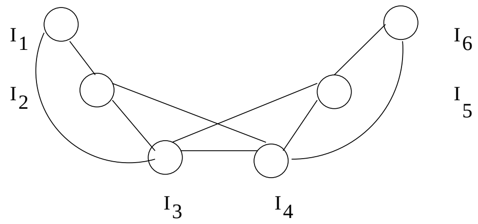

Figure 7.4: The graph G'_1

We now give another characterization of $PUIGs$ that will be useful in devising an algorithm for the best cluster on these graphs.

Lemma 7.2 *Given a UIG $G = (V, E)$. The following are equivalent.*
(a) G is a PUIG.
(b) The subgraph of G' induced by the dominating areas is a simple path. Furthermore every non-dominating area I_i is adjacent to at most two dominating areas I_a and I_b, such that $I_a < I_i < I_b$.

Proof :

$(a) \to (b)$: Let G' have no induced subgraph isomorphic to G'_1 and assume that the dominating areas do not induce a simple path, then either

(i) they induce more than one connected component, or

(ii) they induce an acyclic connected graph (i.e. tree) that is not a path, or

(iii) they induce a connected graph that contains a cycle.

The first two cases are contradictions since they imply respectively that the graph is disconnected (by the definition of dominating area) or not unit interval.

Consider the last case. Since G is chordal then any cycle in G' is of length three. Let I_a, I_b, I_c be three dominating areas that induce a cycle, such that $I_a < I_b < I_c$. By the definition of a dominating area there is an area I_{a1} in $N(I_a) - N(I_b)$, therefore $I_{a1} < I_a < I_b < I_c$ (since G is a UIG). Also I_b has to have neighbors in $N(I_b) - N(I_a)$ and in $N(I_b) - N(I_c)$, therefore there are areas I_{b1} and I_{b2} adjacent to I_b, such that $I_c < I_{b1}$ (so I_{b1} is not adjacent to I_a), and $I_{a1} < I_{b2} < I_a$ (so I_b is not adjacent to I_{a1}). But now there is a subgraph isomorphic to G'_1, induced by the set of areas $\{I_{a1}, I_{b2}, I_a, I_b, I_c, I_{b1}\}$, where the isomorphism function g is such that, $g(I_{a1}) = I_1$, $g(I_{b2}) = I_2$, $g(I_a) = I_3$, $g(I_b) = I_4$, $g(I_c) = I_5$, and $g(I_{b1}) = I_6$.

Therefore the subgraph of G' induced by the dominating areas is a simple path. If there is a non-dominating area I_i that is adjacent to more than two dominating areas: I_a, I_b, and

I_c, $I_a < I_b < I_c$, then since these three dominating areas induce a simple path and there is an area that dominates I_i, there is a dominating area I_d adjacent to all three areas I_a, I_b, and I_c. Then it must be true that $d = b$, i.e. I_b dominates I_i, otherwise I_a, I_b, and I_d are totally connected. Suppose $I_b < I_i < I_c$ (note that I_i cannot be greater than I_c otherwise I_a is adjacent to I_c). There is an area I_{b1} in $N(I_b) - N(I_i)$, therefore $I_{b1} < I_a$ and is adjacent to I_a. Also there is an area I_{a1} in $N(I_a) - N(I_b)$, therefore $I_{a1} < I_{b1}$. But now there is a subgraph isomorphic to G_1', induced by the set of areas $\{I_{a1}, I_{b1}, I_a, I_b, I_i, I_c\}$, where the isomorphism function is g, $g(I_{a1}) = I_1$, $g(I_{b1}) = I_2$, $g(I_a) = I_3$, $g(I_b) = I_4$, $g(I_i) = I_5$, and $g(I_c) = I_6$, a contradiction. Therefore every non-dominating area is adjacent to at most two dominating areas.

Suppose I_i is adjacent to I_a and I_b such that $I_a < I_b < I_i$. Then there is a dominating area I_c, $I_c > I_i$ (otherwise I_a dominates I_b). But now I_i is adjacent to I_c as well, a contradiction.

$(b) \to (a)$ **:** If the dominating areas satisfy the claim but G' has an induced subgraph G_a' isomorphic to G_1', then assume vertices in G_a' are numbered as in G_1'.

No area in G_a' dominates I_3 or I_4. Therefore either I_3 and I_4 are dominating areas in G' or there is an area that dominates each. If I_3 and I_4 are dominating areas then I_2 cannot be a dominating area (since I_2, I_3, and I_4 do not induce a simple path). If I_2 is a non-dominating area then since it is adjacent to both I_3 and I_4 it has to be such that $I_3 < I_2 < I_4$, but now I_2 has to be adjacent to I_5, a contradiction.

If I_3 is dominated by I_{i3} then again I_2 and I_5 cannot be between I_{i3} and I_4 or between I_{i3} and an area that dominates I_4 (otherwise they will be adjacent), a contradiction. ∎

Definition 7.2: *The connecting path, P_c,* is the path induced by the dominating areas in a PUIG.

Pictorially we will represent a *PUIG* in a *Venn* diagram with circles or ellipses representing maximal cliques, in increasing order from left to right. Dominating areas will be those areas in the intersection of circles; for example see figure 7.5 where I_4, I_{10}, and I_{15} are the only dominating areas. Note that using this representation of a *PUIG*, if two maximal cliques, each represented by an ellipse, intersect, then it must be true that all areas in both cliques are dominated by the same area; for example in figure 7.5 the maximal cliques containing I_3 and I_8 cannot intersect.

7.4 Best Cluster on *PUIGs*

In this section we examine some useful properties of a best cluster. First some definitions are due.

7.4.1 Definitions

Let $G = (V, E)$ be a $PUIG$, $G' = (\mathcal{I}, \mathcal{E})$ be its prime graph. Let C^*_k be a best k-cluster in G. Define the following.

The *weight* of an area I_i $w(I_i) = |I_i \bigcap C^*_k|$.

With respect to C^*_k each area I_i is either: *empty* if $w(I_i) = 0$, *full* if $w(I_i) = |I_i| > 0$, *connecting* if $w(I_i) = 1$ and $C^*_k - I_i$ is disconnected, or *partial* if $w(I_i) < |I_i|$ and $C^*_k - \{v_j\}$ is connected for any $v_j \in I_i$.

Figure 7.5 shows a cluster with its area types.

$C^*(a, b, k)$ is a best k-cluster that starts at I_a and ends at I_b, i.e. $\forall I_i < I_a, |I_i \bigcap C^*(a, b, k)| = \phi$, $\forall I_i > I_b, |I_i \bigcap C^*(a, b, k)| = \phi$, $|I_a \bigcap C^*(a, b, k)| \neq \phi$, and $|I_b \bigcap C^*(a, b, k)| \neq \phi$.

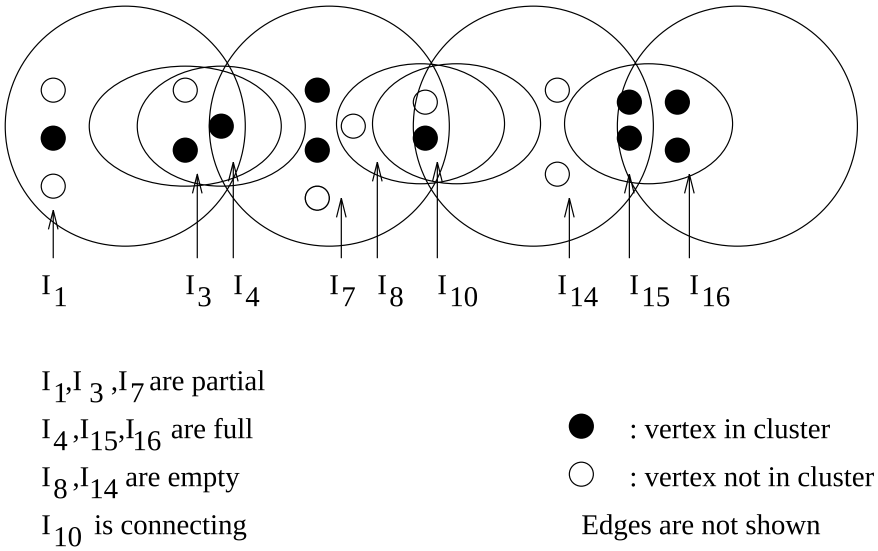

Figure 7.5: A cluster and its different area types

$C^*(a, b, k)$ is one of the following : *type A* if it consists of full or empty areas only, *type B* if it consists of full, empty, or partial areas only, *type AC* if it consists of full, empty, or connecting areas only, and *type BC* otherwise.

We denote by $C^t(a, b, k)$ a best type t k-cluster between I_a and I_b.

Since $C^t(a, b, k)$ might not be unique, we have a set, $\mathcal{C}^t(a, b, k)$, of all type t best k-clusters between I_a and I_b.

For any two sets V_1 and V_2, $V_1, V_2 \subseteq V$, let $E(V_1, V_2)$ be the number of edges with one endpoint in $V_1 - V_2$ and the other in $V_2 - V_1$. Let $E(V_1)$ be the number of edges in the subgraph induced by V_1.

A *block* $B(a, b)$ is the union of consecutive full areas I_i, $a \leq i \leq b$, if as well I_{a-1} and I_{b+1} are empty then $B(a, b)$ is a *maximal block*.

A *bridge* $D(a, b)$ is the union of connecting areas I_i, where $a \leq i \leq b$.

In this section we examine some properties of a best cluster that will help us devise a parallel algorithm for CXE. The final effect of these properties is to enable us to break a best cluster into two smaller (with fewer vertices) best clusters. Lemma 7.3 shows that full areas are always consecutive and form blocks such that any maximal block contains at least one dominating area. As a result, a type A best cluster consists of a sequence of maximal blocks. This property enables us to build any type A best cluster with l maximal blocks from two type A best clusters with fewer than l maximal blocks (lemma 7.4). For types B and BC clusters we prove that there is at most one partial area (lemma 7.5), thus computationally we have to try only $\mathcal{O}(n)$ possibilities for a partial area. For types AC and BC clusters we prove that removing one bridge results into two best clusters of smaller cardinality and fewer bridges (lemma 7.6 and lemma 7.7).

7.4.2 Properties of Best Cluster

We now prove that there is a best cluster C_k^* satisfying the following properties. Let C_k^* be a best k-cluster and suppose that it starts at I_a and ends at I_b, that is $C_k^* = C^*(a, b, k)$.

Lemma 7.3 (Maximal Blocks) *There is a best cluster such that for all areas I_i and I_j, $a \leq i, j \leq b$, if I_i is non-empty and I_i is dominated by I_j, then I_j is full.*

Proof : Let I_j dominate I_i, and both be non-empty. Let $v_a \in I_i \bigcap C_k^*$ and $v_b \in I_j - C_k^*$. Since I_j dominates I_i, therefore $N[v_a] \subset N[v_b]$. Subsequently $(N[v_a] \bigcap C_k^*) \subset (N[v_b] \bigcap C_k^*)$. Thus replacing v_a by v_b in C_k^* gives a cluster with at least as many edges. Therefore we may move vertices from I_i to I_j until I_j becomes full or I_i becomes empty. Since I_j dominates I_i then the cluster will remain connected. ■

Corollary 7.1 *Let $B(i, j)$ be a maximal block in C_k^* then $B(i, j)$ contains at least one dominating area.*

Proof : Follows from the previous lemma. Choose a cluster that satisfies that lemma.

Suppose $B(i, j)$ is a maximal block in C_k^* that does not contain a dominating area. Let I_{i^*} and I_{j^*} be the closest dominating areas to $B(i, j)$ such that $I_{i^*} < I_i$ and $I_{j^*} > I_j$. At least one of I_{i^*} and I_{j^*} exists. Since G is a $PUIG$ then I_i is dominated by at least one of I_{i^*} and I_{j^*}. Suppose I_{i^*} dominates I_i. Therefore I_i is full by the previous lemma. Now I_i has no neighbor larger than I_{j^*} (by the definition of a dominating area and by lemma 7.2 part (b)). Consider area I_{i-1}. I_{i-1} is empty, but all neighbors of I_i that are smaller than I_{i^*} are also neighbors of I_{i-1} (since G is a UIG). Since, as well, I_{i-1} is adjacent to all neighbors of I_i that are between I_{i^*} and I_{j^*} (inclusive), therefore I_{i-1} dominates I_i. This implies that C_k^* does not satisfy the previous lemma, a contradiction.

This property means that vertices of C_k^* cluster around vertices of P_c (see figure 7.6). ■

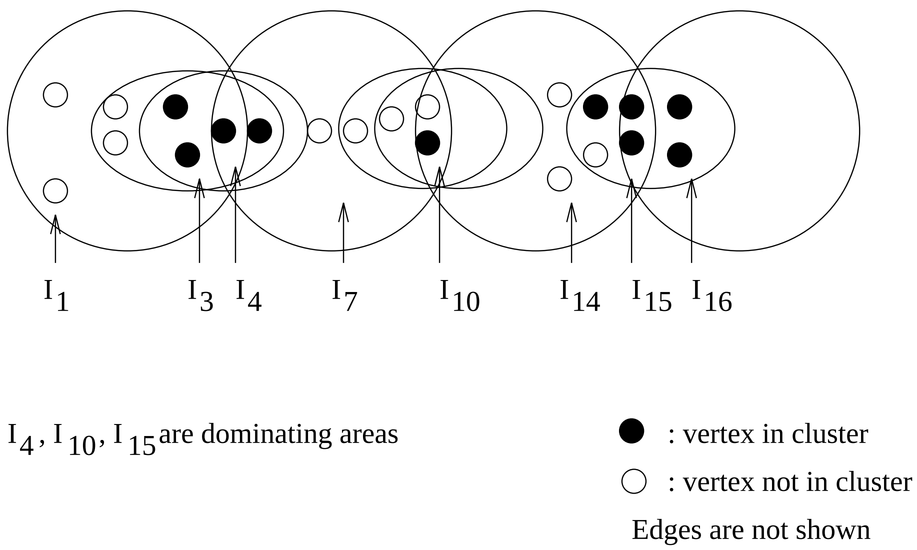

Figure 7.6: Full areas cluster around P^c

Corollary 7.2 *Let $D(i,j)$ be a bridge in C_k^* then all areas I in $D(i,j)$ are in P^c.*

Proof : Follows directly from the previous corollary; consider $D(i,j)$ to be the union of maximal blocks each containing one area that has unit capacity. ∎

Lemma 7.4 (Type A Clusters) *Let C_k^* be a type A best cluster, $C_k^* = C^A(a,b,k)$. Let $T_R = C^A(a,b,k) - B(a,j)$, where $B(a,j)$ is the maximal block in C_k^* that starts at I_a. Then there is a type A best cluster C_k^* such that T_R is in $\mathcal{C}^A(i,b,k1)$, where I_i is the first full area of C_k^* that follows I_j, and $k1 = k - |B(a,j)|$.*

Proof : Let C_k^* satisfy lemma 7.3. Assume the claim is not true, then there is a cluster $C_R' \in \mathcal{C}^A(i,b,k1)$ such that C_R' is a best $k1$-cluster between I_i and I_b and it satisfies lemma 7.3.

Therefore $E(C_R') > E(T_R)$, but $E(C_R', B(a,j)) < E(T_R, B(a,j))$. Let $I_{i'}$ be the area with the smallest index in $B(a,j)$ that is adjacent to I_i. Let $I_{j'}$ be the area with the largest index in C_R' that is adjacent to $I_{i'}$.

Since C_k^* satisfies lemma 7.3 and G is a $PUIG$, therefore $I_{i'}$ is the largest index dominating area in $B(a,j)$ (see figure 7.7). Likewise, C_R' satisfies lemma 7.3, hence $I_{j'}$ is the smallest index dominating area in C_R'. The areas $I_i, \ldots, I_{j'}$ form a block (by lemma 7.3), and $E(C_R', B(a,j)) = E(B(i,j'), B(i',j))$ (since G is a PUIG). If T_R does not contain empty areas between I_i and $I_{j'}$ (inclusively), then $E(C_R', B(a,j)) = E(T_R, B(a,j))$. Otherwise T_R contains some empty area between I_i and $I_{j'}$, a contradiction to our choice of $B(a,j)$ or to the fact that C_k^* satisfies lemma 7.3. ∎

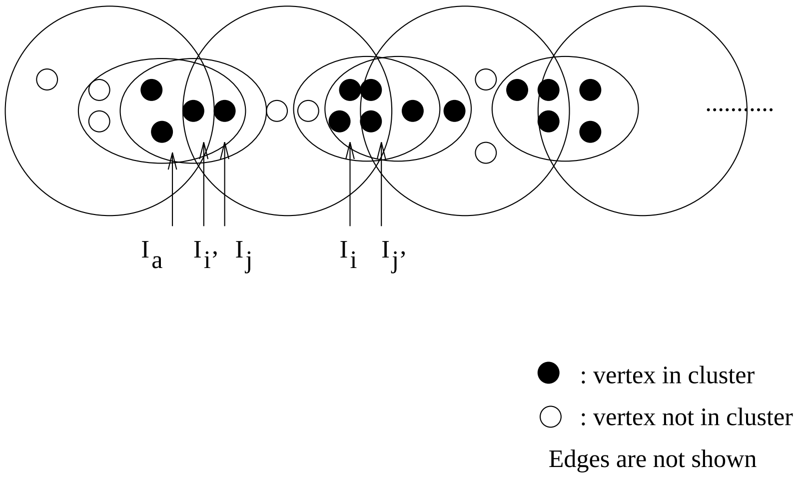

Figure 7.7: Part of a type A cluster

Lemma 7.5 (Types B and BC Clusters) *There is a best cluster C_k^* that has at most one partial area.*

Proof : Assume there are two areas I_l and I_o that are both partial. For any $v_i \in I_l \bigcap C_k^*$ and $v_j \in I_o \bigcap C_k^*$, let $n_i = |N(v_i) \bigcap C_k^*|$ and $n_j = |N(v_j) \bigcap C_k^*|$. Suppose $n_i \geq n_j$. Let $v' \in I_l - C_k^*$ and $v'' \in I_o \bigcap C_k^*$. Thus $N(v') \bigcap C_k^* = N[v_i] \bigcap C_k^*$, and $|N(v') \bigcap C_k^*| > n_j$. Therefore substituting v' for v'' in C_k^* gives a cluster with at least as many edges. Notice that by substituting v' for v'', we increase the weight of I_l in C_k^* and decrease that of I_o. Therefore we further increase $|N(v_i) \bigcap C_k^*|$, for all v_i in I_l, and further decrease $|N(v_j) \bigcap C_k^*|$ for all v_j in I_o. We can repeat this process until either I_o is empty, I_l is full, or I_o is a connecting area. ■

> **Notation:** We will use the notation $C^B(a, b, k/jj, w)$ or $C^{BC}(a, b, k/jj, w)$ to denote the cluster having a partial area I_{jj} of weight w.

Lemma 7.6 (Type AC Clusters) *Let C_k^* be a type AC cluster. Let $D(i1, l1)$ be a bridge in C_k^*. Let I_i and I_l be the previous and the following full areas. Let $T_L = C_k^* \bigcap (\bigcup_{k' \leq i} I_{k'})$, $T_R = C_k^* \bigcap (\bigcup_{k' \geq l} I_{k'})$, $n_L = |T_L|$, and $n_R = |T_R|$. Then T_L is in $\mathcal{C}^{AC}(a, i, n_L)$ and T_R is in $\mathcal{C}^{AC}(l, b, n_R)$.*

Proof : Let $S_R = C_k^* - T_L$. By corollary 7.2 all $I_{i1}, \ldots\ldots I_{l1}$ are in P_c. Assume the claim is false. Therefore there is a cluster $C_L' \in \mathcal{C}^{AC}(a, i, n_L)$ that satisfies lemma 7.3 such that $E(C_L') > E(T_L)$, but $E(C_L', S_R) < E(T_L, S_R)$. Since C_L' is a type AC best cluster, then by lemma 7.3 and the fact that G is a $PUIG$, it follows that $E(C_L', S_R) = E(C_L', I_{i1}) = |I_{i1}||N(I_{i1}, C_L')| = |B(i, j)|$, where I_j is the smallest index dominating area smaller than I_{i1}

(since G is a $PUIG$, see figure 7.8). But since T_L ends at I_i as well and C_k^* satisfies lemma 7.3, then $N(I_{i1}, T_L) = I_i \bigcup \ldots \bigcup I_j$. This implies $E(C_L^{'}, S_R) = E(T_L, S_R)$; a contradiction.

Similarly we can prove that T_R is in $\mathcal{C}^{AC}(l, b, n_R)$. ■

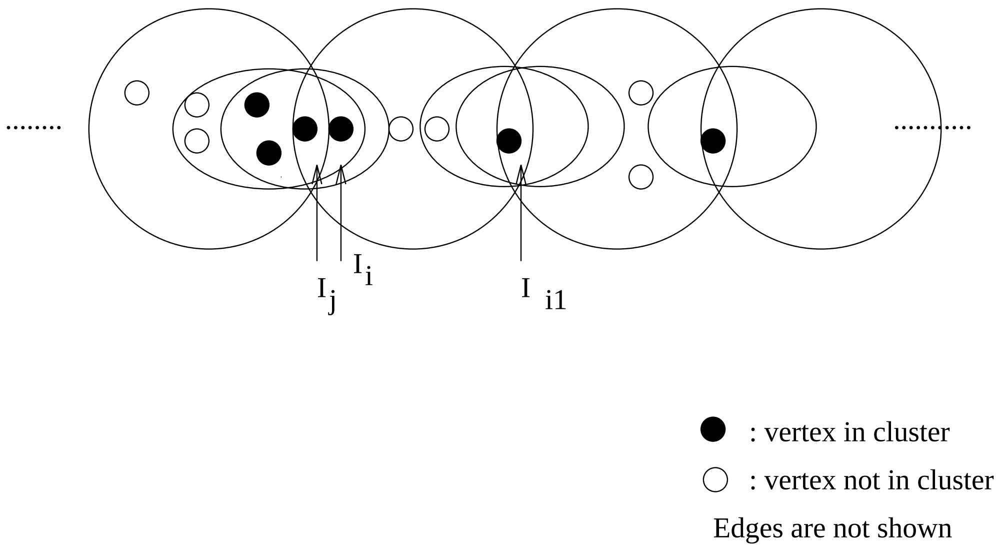

Figure 7.8: Part of a type AC cluster

Lemma 7.7 (Type BC Clusters) *Let C_k^* be a type BC cluster. Let $D(i1, l1)$ be a bridge. Let I_i and I_l be the previous and the following non-empty areas. $C_k^* = C^{BC}(a, b, k/jj, w)$ for some a, b, jj, w, $jj \leq i$ or $jj \geq l$ (by lemma 7.5). Let $T_L = C_k^* \bigcap (\bigcup_{k' \leq i} I_{k'})$, $T_R = C_k^* \bigcap (\bigcup_{k' \geq l} I_{k'})$, $n_L = |T_L|$, and $n_R = |T_R|$. Then T_L is in $\mathcal{C}^{BC}(a, i, n_L)$ and T_R is in $\mathcal{C}^{AC}(l, b, n_R)$, or T_L is in $\mathcal{C}^{AC}(a, i, n_L)$ and T_R is in $\mathcal{C}^{BC}(l, b, n_R)$.*

Proof : Similar to lemma 7.6 (consider $c(I_{jj})$ to be equal to w). ■

7.5 Parallel Algorithms to Find the Best k-cluster on $PUIGs$

In this section we present efficient parallel algorithms to find C_k^* when the input graph is a $PUIG$. The input is assumed to be the prime graph $G' = (\mathcal{I}, \mathcal{E})$.

The algorithm has a procedure for each cluster type. The main procedure is that for type A clusters. In that procedure, after the ith iteration of the main loop all best type A clusters that consist of less than or equal to 2^i blocks are correctly computed. Since we have at most $p \leq n$ blocks then this loop requires at most $\mathcal{O}(\log n)$ time. The procedure for type B clusters merely calls that for type A for all possibilities of a partial area. The main loops for the procedures for types AC and BC are similar to that of type A procedure; with each iteration the maximum number of bridges among all correctly calculated best clusters is doubled.

In all these procedures we use the operator "max" to obtain the cluster with the maximum number of edges. We assume the input to the algorithm is the prime graph obtained by algorithm 7.1. We use Λ to denote *undefined* clusters.

Algorithm 7.2 Algorithm BEST-CLUSTER(G', k, C)

Input: $G' = (\mathcal{I}, \mathcal{E})$, and k, *where k is the cardinality of the best cluster*
Output: $C_k^* = C^*(1, p, k)$.

begin

1. *Call TYPE-A*$(G', k, 1, p, C^A)$
2. *Call TYPE-B*$(G', k, 1, p, C^B)$
3. *Call TYPE-AC*$(G', k, 1, p, C^{AC})$
4. *Call TYPE-BC*$(G', k, 1, p, C^{BC})$
5. $C_k^* = \max(C^A, C^B, C^{AC}, C^{BC})$

end

Procedure TYPE-A(G', k, f, l, C)

Input: $G' = (\mathcal{I}, \mathcal{E})$, *integer k.*
Output: $C = C^A(f, l, k)$, *a cardinality k type A best cluster the starts at I_f and ends at I_l.*

begin

1. **for all** $i, j, k_1,\ f \le i \le j \le l,\ k_1 \le k$, **do in parallel**
   $C^A(i, j, k_1) = \Lambda$
   endfor
2. **for all** $i, j, f \le i \le j \le l$ **do in parallel**
   $B(i, j) = \bigcup_{i \le o \le j} I_o$
   $C^A(i, j, |B(i, j)|) = B(i, j)$
   endfor
3. **repeat** $\lceil \log n \rceil$ **times**
   (a) **for all** $i, j, k_1, f \le i \le j \le l, k_1 \le k$ **do in parallel**
   $C^A(i, j, k_1) = \max_{j1, j2, l_1}(C^A(i, j1, l_1) \bigcup C^A(j2, j, k_1 - l_1))$
   (where $i \le j1 < j2 \le j$, I_{j2} is adjacent to I_{j1})
   endfor
4. $C = C^A(f, l, k)$

end

Procedure TYPE-B(G', k, f, l, C)

Input: $G' = (\mathcal{I}, \mathcal{E})$, *integer* k.
Output: $C = C^B(f,l,k)$, *a cardinality* k *type B best cluster the starts at* I_f *and ends at* I_l.

begin

1. **for all** $jj, w,\ f \leq jj \leq l,\ w < c(I_{jj})$ **do in parallel**
   $G'_a = G'$
   set $c(I_{jj})$ **to** w **in** G'_a
   Call TYPE-A(G'_a, k, f, l, C)
   $C^B(f,l,k/jj,w) = C$
   endfor
2. $C^B(f,l,k) = \max_{jj,w} C^B(f,l,k/jj,w)$ **endfor**
3. $C = C^B(f,l,k)$

end

Procedure TYPE-AC(G', k, f, l, C)

Input: $G' = (\mathcal{I}, \mathcal{E})$, *integer* k.
Output: $C = C^{AC}(f,l,k)$, *a cardinality* k *type AC best cluster the starts at* I_f *and ends at* I_l.

begin

1. **for all** $i, j, k_1, f \leq i \leq j \leq l, k_1 \leq k$ **do in parallel**
   $C^{AC}(i,j,k_1) = C^A(i,j,k_1)$
   endfor
2. **for all** i, j **do in parallel**
   $D(i,j) = \bigcup\{v_{i1} : v_{i1} \in I_{j1}, i \leq j1 \leq j, I_{j1} \in P^c\}$
   endfor
3. **repeat** $\lceil \log n \rceil$ *times*
   (a) **for all** $i, j, k_1, f \leq i \leq j \leq l, k_1 \leq k$ **do in parallel**
   $C_1 = \max_{i1,i2,k11}(C^{AC}(i,i1,k11) \bigcup D(i11,i1l) \bigcup C^{AC}(i2,j,k_1 - k11 - |D(i11,i1l)|))$
   (where $i \leq i1 < i11 \leq i1l < i2 \leq j1$, I_{i1} *is adjacent to* I_{i11}, I_{i1l} *is adjacent to* I_{i2}*)*
   $C^{AC}(i,j,k_1) = \max(C_1, C^{AC}(i,j,k_1))$
   endfor
4. $C = C^{AC}(f,l,k)$

end

Procedure TYPE-BC(G', k, f, l, C)

Input: $G^{'} = (\mathcal{I}, \mathcal{E})$, *integer* k.
Output: $C = C^{BC}(f, l, k)$, *a cardinality* k *type BC best cluster the starts at* I_f *and ends at* I_l.

begin

1. **for all** $i, j, k_1, jj, w, f \leq i \leq jj \leq j \leq l, k_1 \leq k$ **do in parallel**
   $C^{BC}(i, j, k_1/jj, w) = C^B(i, j, k_1/jj, w)$
   endfor
2. **for all** i, j **do in parallel**
   $D(i, j) = \bigcup\{v_{i1} : v_{i1} \in I_{j1}, i \leq j1 \leq j, I_{j1} \in P^c, v_{i1} \leq v_{l1}, \forall v_{l1} \in I_{j1}\}$
   endfor
3. **repeat** $\lceil \log n \rceil$ *times*
   (a) **for all** $i, j, k_1, f \leq i \leq j \leq l, k_1 \leq k$ **do in parallel**
      i. **for all** $jj, w,\ i \leq jj \leq j,\ w < c(I_{jj})$
         $C_1 = \max(\max_{i1,i2,k11}(C^{BC}(i, i1, k11/jj, w) \bigcup D(i11, i1l))$
         $\bigcup C^{AC}(i, i1, k_1 - k11 - |D(i11, i1l)|)), (\max_{i1,i2,k11}(C^{AC}(i2, j, k11) \bigcup D(i11, i1l))$
         $\bigcup C^{BC}(i2, j, k_1 - k11 - |D(i11, i1l)|))$
         (where $i \leq i1 < i11 \leq i1l < i2 \leq j1$*,* I_{i1} *is adjacent to* I_{i11}*,*
         and I_{i1l} *is adjacent to* I_{i2}*)*
         $C^{BC}(i, j, k_1/jj, w) = \max(C_1, C^{BC}(i, j, k_1/jj, w))$
         endfor
      ii. $C^{BC}(i, j, k_1) = \max_{jj,w} C^{BC}(i, j, k_1/jj, w)$
      endfor
4. $C = C^{BC}(f, l, k)$

end

Theorem 7.1 *Algorithm 7.2 computes the number of edges in the best* k*-cluster in* $\mathcal{O}(\log^2 n)$ *time using* $\mathcal{O}(n^7 k^4)$ *processors.*

Proof : Since any cluster is either of type A, B, AC, or BC, then it is sufficient to prove the correctness and find the complexity of each of the four procedures individually.

We first prove that the procedure for type A clusters is correct. We prove this by induction on the number of blocks b, $b \leq 2^l$, in C_k^*. If $l = 0$ then the procedure is correct since each cluster is initially set equal to a block.

Assume it is correct for all clusters having 2^{l-1} or fewer blocks. Let C_k^* have 2^l or fewer blocks, and suppose that $C_k^* = C^A(i, j, k)$. There is a full area $j1$ that ends a maximal block (corollary 7.1), such that the number of maximal blocks in C_k^* that contain any area smaller than or equal to $j1$, is smaller than or equal to 2^{l-1}, and the number of maximal blocks that contain any area larger than $j1$, is smaller than or equal to 2^{l-1}.

Let T_L contain all vertices in C_k^* that are in areas smaller than or equal to $j1$, and let T_R contain the rest of the vertices in C_k^*. By lemma 7.4 and by the associativity of the union operator, $T_L \in \mathcal{C}^A(i, j1, k1)$. Similarly T_R is a best type A cluster.

The correctness of the procedure for type B follows from lemma 7.5 and the correctness of the procedure for type A.

For type AC (or type BC) clusters, we prove the correctness of the procedures by induction on the number of bridges, $b \leq 2^l$, in the cluster. If $l = 0$ then by lemma 7.4 there are best clusters C_R and C_L that are, in this case, type A clusters. Since type A clusters are correctly computed, therefore the procedure performs correctly for the base case. n

Assume it is correct for all clusters having less than or equal to 2^{l-1} bridges. Let C_k^* have $b \leq 2^l$ bridges. Then there is a bridge in C_k^* whose removal results in two best clusters C_R and C_L, each of which is connected but their union is not (as in lemma 7.4); and each of these clusters has less than or equal to 2^{l-1} bridges. Therefore by the induction hypothesis they are correctly computed.

We now compute the complexity of the algorithms. Suppose we are given the prime graph of G, G'. For every area I_i, min_i is the smallest index vertex in I_i, and let max_i be the largest index vertex in I_i. Both min_i and max_i may be efficiently obtained in parallel from the prime graph. To compute min_i and max_i, for all i, we need at most $\mathcal{O}(n^2)$ processors and $\mathcal{O}(\log n)$ time. We now show how to efficiently compute some measures pertaining to the algorithms. Suppose that for all areas I_i, $f(I_i)$, $l(I_i)$, min_i, and max_i have already been computed.

(a) To obtain a list of dominating areas $\{I_1^c, I_2^c, \ldots\}$, for each area I_i compute $f(I_i)$ and $l(I_i)$. Then $I_i \in P^c$, if and only if there is no area I_j, $j \neq i$, such that $f(I_j) \leq f(I_i) < l(I_i) \leq l(I_j)$. Therefore at most $\mathcal{O}(n^2)$ processors and constant time is needed to find P^c.

(b) For each $B(i,j)$ it is sufficient to store i and j. Then $|B(i,j)| = max_j - min_i + 1$. Therefore constant time and one processor are required for this step.

(c) The number of edges between any two type A clusters $C^A(a, i, k_1)$ and $C^A(j, b, k_2)$, $a \leq i < j \leq b$ is $(max_i - min_{i^*} + 1) * (max_{j^*} - min_j + 1)$, where i^* is the index of the first dominating area in $C^A(a, i, k_1)$ that precedes i, and j^* is the index of the first dominating area in $C^A(j, b, k_2)$ that succeeds j.

(d) For each $C^t(a, b, k)$ it is sufficient to store k, min_a, max_{j^*}, max_b, and min_{i^*}, where j^* is the index of the first dominating area to succeed I_a, and i^* is the index of the first dominating area to precede I_b.

(e) For type B clusters, if area I_{jj} has capacity w, then let $min_{jj} = max_{jj} - w + 1$ in the graph G_a' (see the procedure for type B clusters).

(f) It is sufficient to store i and j for $D(i,j)$, $|D(i,j)|$ = number of dominating areas between i and j inclusive, and the number of edges induced by $D(i,j)$ equals $|D(i,j)| - 1$.

Therefore to compute the number of edges in a cluster or the cardinality of a cluster, at most $\mathcal{O}(n^2)$ processors and $\mathcal{O}(\log n)$ time is needed.

The complexity of the procedure for type A clusters is $\mathcal{O}(n^4k^2)$ processors and $\mathcal{O}(\log^2 n)$ time. The explanation for this follows. The first loop requires $\mathcal{O}(n^3)$ processors. The second loop requires a processor for each pair (i,j), and constant time to compute $B(i,j)$, $|B(i,j)|$, and $C^A(i,j,|B(i,j)|)$ (see observations (b) and (c) above). Therefore that loop needs a total of $\mathcal{O}(n^2)$ processors and $\mathcal{O}(1)$ time. The outer loop (step 3) requires $\mathcal{O}(\log n)$ time. The inner loop (step 3.a) requires $\mathcal{O}(n^2k)$ processors for each triplet (i,j,k_1). To compute the union of two clusters one processor and constant time are needed (see observation (d) above). To compute the maximum $\mathcal{O}(n^2k)$ processors and $\mathcal{O}(\log n)$ time is needed. The complexity of the procedure of type B clusters is $\mathcal{O}(n^5k^3)$ processors and $\mathcal{O}(\log^2 n)$ time.

The complexity of the algorithm is that of the procedure for type BC clusters. Specifically its the complexity for the loop that calls TYPE-B procedure. In total that loop requires $\mathcal{O}(n^7k^3)$ processors and $\mathcal{O}(\log^2 n)$ time. ∎

Chapter 8

Discussion and Conclusions

We are finally ready to conclude this thesis. We start by giving a brief account of its contribution.

8.1 Summary of Results

The problems studied in this thesis may be classified as clustering problems. We studied two problems. On the one hand we sought a partition of the vertices of a graph. On the other hand we sought a single subgraph of the given graph.

The first problem we studied is that of partitioning vertices of a graph into the minimum number of sets so that the partition satisfies certain requirements on the diameters of the induced subgraphs. We called this problem $PVMF$. For example the sets may be required to have a diameter less than or equal to a certain integer B. We called this problem PVD. We note that PVD is itself a generalization of the well known problem of partitioning vertices into cliques (PVC), obtained by setting B to one. Or, as another example for a problem in $PVMF$, the sum of the diameters of the subgraphs may be required to be bounded above by an integer B.

Our contribution to that problem may be summarized as follows.

(a) We introduced *monotone diameter functions* and showed how this enabled us to group the aforementioned problems, and others, into one problem, namely $PVMF$.

(b) We proved that the special case PVD is *NP*-complete on bipartite graphs. We also showed PVD to be *NP*-complete on chordal graphs, specifically on split an undirected path graphs (also shown in [26], and implied by the results in [21], [9] and [12]).

These results are in contrast with the fact that PVD has a linear time solution on trees [31], which are both chordal and bipartite. The *NP*-completeness result on split graphs is for $B = 2$. It is interesting to note that for $B = 1$ the problem PVD is easy for split graphs, and for $B \geq 3$ it is trivial since split graphs have diameters less than or equal to three.

The *NP*-completeness result on bipartite graphs is for all fixed values of B greater than one. We should note that PVD with $B = 1$ has a polynomial time solution for bipartite graphs. The *NP*-completeness result for $B = 2$ implies that it is *NP*-complete to partition vertices of bipartite graphs into complete connected bipartite graphs, a result which has not been proven prior to this thesis.

(c) In contrast with these *NP*-completeness results, we showed how to solve PVD sequentially in linear time on bipartite permutation and interval graphs, the first being a subset of bipartite graphs and the second being a subset of chordal graphs. These algorithms are optimal and the results extend the linear time solvability of PVC (partitioning into cliques) on these two graph families.

(d) We also presented efficient parallel algorithms for $PVMF$ on interval and biconvex graphs.

(e) Our study of $PVMF$ on biconvex graphs led us to study properties of this family in greater detail than we first anticipated. We proved that vertices of biconvex graphs may be ordered so that in addition to biconvexity, they also possess, what we called, a *biconvex straight ordering* (or biconvex S-ordering for short). This ordering is a generalization of the strong ordering achievable for bipartite permutation graphs, which are a proper subset of biconvex graphs. We presented an efficient parallel algorithm to reorder vertices of a biconvex graphs so they have this new ordering. Using the new ordering, we were able to prove numerous properties about distances and diameters in these graphs, that we later used to find a special optimal partition.

The second problem we studied was that of finding a subgraph having a number of vertices bounded above by an integer k that induces the maximum possible number of edges. We presented an efficient parallel algorithm for this problem on a class of graphs that is a subset of unit interval graphs. We characterized this class in terms of a forbidden subgraph. We also made use of the notion of a *prime graph* where vertices having the same neighborhood set in the original graph are collapsed into one vertex in the prime graph, and used that graph as input to the parallel algorithm. This algorithm is the first parallel algorithm for CXE, and is one of few other algorithms for CXE (other algorithms for CXE are on cographs [21] and partial h-trees for fixed h [54]).

We now discuss possible extensions to these results.

8.2 Suggestions for Future Research

Extensions to these results may flow in any of a number of directions. We discuss some of these directions.

Other Related Problems:

An interesting question arises about the 'duals' of problems $PVMF$ and CXE. We exemplify what we mean by duals. Consider, for instance, the problem: Given a graph G, does G have an induced subgraph on k or more vertices that has diameter less than or equal to B? Or, the problem: Given a graph G, does G have a partition of its vertex set into subgraphs of bounded size such that each induced subgraph has the maximum possible number of edges? We are unaware of any results in the literature on the first problem. However the second problem has been studied on trees, see [2], [63], and [6].

When an application calls for clustering vertices of a graph, the problem might allow vertices to be in more than one set. That is, sets of the optimal partition need not be disjoint. We saw that for $PVMF$ we were able to find a partition of disjoint sets for interval and biconvex graphs. The sets in the partition for biconvex graphs are disjoint but the ranges of the sets are not disjoint. Certainly if we do not have any restrictions on the number of vertices in the intersections of set pairs, then there will be no difficulty whatsoever in devising efficient clustering algorithms. However a reasonable assumption to make is that at most k_1 sets mutually intersect, and that the number of vertices in the intersection of any k_1 sets is less than or equal to k_2. Hence, a special case of that clustering problem would be that of clustering into non-overlapping sets, with $k_1 = 1$ and $k_2 = 0$. Another special case with $k_1 = 2$ and $k_2 = 1$, would be to find a partition where at most two sets mutually overlap and they share no more than one vertex. It seems that such a relaxation might render some hard problems easier to solve.

We found how our introduction of monotone diameter functions unified a few problems under one umbrella, and one algorithm was sufficient to solve these problems. Such generalizations are very beneficial considering the volume of problems and algorithms that exist in the literature; are there similar generalizations to some of the existing problems, or are there any relations between seemingly different problems? For example, we showed the equivalence between maximum matching and PVD with $B = 1$ when the input graph is bipartite [31], and the equivalence of the minimum dominating set problem and PVD with $B = 2$ when the input graph is chordal. The latter equivalence does not extend to bipartite graphs (for example consider $K_{2,2}$, it does not contain a dominating vertex). However we can generalize this relation differently. When $B = 3$ it is easy to see that a solution for PVD on bipartite graphs, offers a solution to the *minimum mixed dominating set* problem, a problem whose solution is a dominating set that consists of either vertices or edges. In the other direction a solution to the minimum mixed dominating set problem, may be extended to a solution for PVD with $B = 3$ on bipartite graphs. It would be interesting to investigate generalizations for higher values of B.

An interesting direction to follows is to substitute the diameter in the definition of $f(\mathcal{Z})$ by other meaningful measures, and study the applicability of our algorithms to the new function.

For instance, the average distance in a cluster might be a good candidate.

Another related problem is to find a partition when the edges of a graph have weights. Such a problem appears more practical than its unweighted counterpart. For problem CXE, this generalization is studied in [70], on trees. We are unaware of such generalizations for PVD.

Extension of Results to Other Graph Families:

Another direction that one might pursue, is to find how far the solvability of the problems $PVMF$ and CXE can be extended to narrow the gap between the *NP*-complete and the polynomial instances.

The solvability of $PVMF$ on interval graphs and its *NP*-completeness on undirected path graphs leaves little space for extension on other chordal graphs that include interval graphs. However one possible path is that between interval and strongly chordal graphs; that path includes directed path graphs as well. We should note that PVD with $B = 2$ is solvable on these graphs (this is implied by the results in [30] and [12]). As well, interval graphs are cocomparability graphs, and thus far there is no *NP*-completeness results for that problem on cocomparability graphs. The fact that we obtained an efficient parallel algorithm for bipartite permutation graphs, which are also a subset of cocomparability graphs, gives hope that $PVMF$ is solvable on cocomparability graphs. This is also motivated by the fact that cocomparability graphs have a special ordering of their vertices of good algorithmic quality. It is also worth mentioning that there are special cases of $PVMF$ whose status is unknown for special classes of cocomparability graphs. For example the status of PVD on permutation graphs is unknown. Permutation graphs are also a good candidate to study since they have a special ordering of the vertices with good algorithmic properties.

We wonder if the algorithms we developed for interval graphs and biconvex graphs work, as they are or with minor modifications, for other families. For instance, can we modify the algorithm for biconvex graphs so it works on convex graphs? That is not readily apparent, since in almost all of our proofs we relied on the fact that the graph has the biconvexity property. Along this direction, we would like to know if there is a 'largest' family of graphs for which there is a non-overlapping optimal partition for the problem $PVMF$, or any of its subproblems. An assertive answer to that question implies the existence of an efficient parallel algorithm for this large family of graphs. For instance is there a theorem of the form: "G has an ordering of its vertices such that G has an optimal non-overlapping partition for problem P if and only if G has property R". For example R may be a specific ordering of the complete graphs of G.

On the negative side, we wonder if other special cases of $PVMF$, besides PVD, are *NP*-complete on bipartite and chordal graphs, for example the problem of partitioning the vertices so that the sum of the diameters of the sets is bounded.

Using Results to Solve Harder Problems:

One always wants to make the best of every positive result obtained. So we wonder whether or not it is possible to use the algorithms obtained on special graphs to devise good approximate algorithms for provably hard problems? For example can we use the algorithm we obtained for $PVMF$ on biconvex graphs to find a good approximate solution for the same problem on bipartite or convex graphs? One approach might be to delete the minimum number of edges from a bipartite graph so the resulting graph is biconvex with a diameter as 'close' as possible to that of the original graph, then apply the algorithm. This certainly appears to be a fruitful research direction if approximate solutions 'close' enough to the exact solution may be obtained.

We would also like to make more use of the new ordering of biconvex graphs (biconvex S-ordering) to solve problems on biconvex graphs. It is very fortunate that that biconvex S-orderings may be obtained efficiently in parallel. We hope that opens the door to other efficient parallel algorithms on biconvex graphs. One of the characteristics of a biconvex graph G that has a biconvex S-ordering is that it contains an induced subgraph G_1 that satisfies the following. G_1 is a strongly ordered bipartite permutation graph, and the neighborhood set of every vertex in G is contained in G_1. This property might suggest that some algorithms for bipartite permutation graphs may be used to solve problems on biconvex graphs.

Other Useful Graph Characterizations:

The straight ordering we obtained for biconvex graphs, has indeed proved itself most useful algorithmically. This ordering, being a generalization of the strong ordering for bipartite permutation graphs, raises the question of whether there are other graph families that have this property. It is not hard to find graphs that have this property but are not chordal bipartite (see figure 8.1).

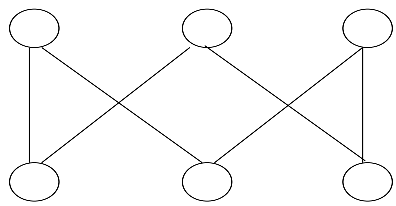

Figure 8.1: A graph that has an S-ordering but is not chordal bipartite

The answer to the above question has further implications. It is known that the adjacency matrix of a chordal bipartite graph is a *totally balanced matrix* (see [29] and [15]), where $A(i,j)$ is an adjacency matrix such that $a(i,j) = 1$ if and only if x_i is adjacent to y_j in $G = (X, Y, E)$, and all other entries are zero. Therefore adjacency matrices of biconvex graphs are those 0-1 matrices that have the consecutive 1's property for both the rows and the columns. They are also totally balanced matrices. The new ordering we obtained implies that they also have an arrangement of their rows and columns so that they are totally balanced, have the consecutive 1's property for both the rows and the columns, and do not contain the following 2x2 matrix:

$$\begin{matrix} 0 & 1 \\ 1 & 0 \end{matrix}.$$

Therefore, do all totally balanced matrices have an arrangement of their rows and columns so they do not contain the above submatrix?

We would also like to further study this ordering to know how unique it is. For example if we collapse all vertices with the same neighborhood set, will we get a unique biconvex S-ordering of the vertices?

There is a relationship between chordal and chordal bipartite graphs that has been investigated in the literature. For instance it is known that clique-vertex graphs of interval graphs are convex graphs [15], and it is easy to see that those of unit interval graphs are BPGs. Similar relations exist between other class of chordal graphs and classes of chordal bipartite graphs.

The similarity of these two families also imposed itself during the course of study of this thesis. During our study of CXE, we found that BPGs have an ordering of their maximal complete bipartite graphs similar to the ordering of the maximal cliques for unit interval graphs. Additionally, it is not hard to see that the parallel algorithm for $PVMF$ on interval graphs is essentially the same as that for bipartite permutation graphs. A question emerges: is there an ordering of the maximal complete graphs of chordal bipartite graphs similar to that of the maximal cliques of chordal graphs (i.e. in a tree)? If the answer is positive then many algorithms on chordal graphs may extend to chordal bipartite graphs, and vice versa.

More Work on CXE:

Problem CXE is one of the problems that has more negative results in the literature than positive results. One hopes the algorithm we developed for CXE can be modified in some way to work for all unit interval graphs, or to work for bipartite permutation graphs. It seems that with few modifications that algorithm might work if we allow the subgraph induced by the dominating areas to be a cycle or a tree, and reserve all other relations between dominating and non-dominating areas, in particular not allow non-dominating areas to be adjacent to more than two dominating areas.

Another possible extension is to consider clusters that are not necessarily connected. It seems that our proofs might need slight modifications to accommodate this case.

Reducing the Complexity of the Algorithms:

Although we showed both $PVMF$ and CXE to belong to the class *FNC*, on special graphs, it remains to determine whether or not the complexities of the algorithms might be reduced. In particular, we would like to reduce the number of processors used by the algorithm for $PVMF$ on biconvex graphs and that of CXE.

Bibliography

[1] N. Abbas and L. Stewart. Partitioning vertices of chordal graphs. *Congressus Numerantium*, 93:7–16, 1993.

[2] E. Agasi, R.I. Becker, and Y. Perl. A shifting algorithm for constrained min-max partition on trees. *Disc. Appl. Math.*, 45:1–28, 1993.

[3] T. Asano, B.K. Bhattacharya, J.M. Keil, and F.F. Yao. Clustering algorithms based on minimum and maximum spanning trees. In *Proc. Fourth Annual Symp. Computational Geometry*, pages 252–257, 1988.

[4] D. Avis. Diameter partitioning. *Disc. and Comp. Geometry*, 1:265–276, 1986.

[5] B.S. Baker. Approximation algorithms for NP-complete problems on planar graphs. In *Proc. 26th Symp. Foundations of Computer Science*, pages 265–273, 1983.

[6] R.I. Becker and S.R. Schach. A shifting algorithm for min-max tree partitioning. *J. ACM*, 29(1):58–67, 1982.

[7] C. Beeri, R. Fagin, D. Maier, and M. Yannakakis. On the desirability of acyclic database schemes. *J. ACM*, 30:479–513, 1983.

[8] F. Berman, D. Johnson, T. Leighton, P.W. Shor, and L. Snyder. Generalized planar matching. *J. Algorithms*, 11:153–184, 1990.

[9] A.A. Bertossi. Dominating sets for split and bipartite graphs. *Infor. Proc. Letters*, 19:37–40, 1984.

[10] A.A. Bertossi and M.A. Bonuccelli. Some parallel algorithms on interval graphs. *Disc. Appl. Math.*, 16:101–111, 1987.

[11] H.L. Bodlaender. Classes of graphs with bounded treewidth. Technical Report TR RUU-CS-86-22, Dept. Computer Science, University of Utrecht, 1986.

[12] K. S. Booth and J. H. Johnson. Dominating sets in chordal graphs. *SIAM J. Comput.*, 11:191–199, 1982.

[13] E. Boros and P.L. Hammer. On clustering problems with connected optima in euclidean spaces. *Disc. Math.*, 75:81–88, 1989.

[14] A. Brandstädt. Monotone partitions of permutations. In *Proc. Colloquium on Algebra, Combinatorics and Logic in Computer Science, Györ, Hungary*, September 1983.

[15] A. Brandstädt. Classes of bipartite graphs related to chordal graphs and domination of one color class by the other. Technical Report N/88/10, Friedrich-Schiller-Universitat Jena, 1988.

[16] A. Brandstädt. The jump number for biconvex graphs and rectangle covers of rectangular regions. Technical Report N/89/01, Friedrich-Schiller-Universitat Jena, 1989.

[17] P. Brucker. On the complexity of clustering problems. In M. Beckman and H.P. Kunzi, editor, *Optimization and Operations Research: Lecture Notes in Economics and Mathematical Systems*, pages 45–54. Springer-Berlin, 1978.

[18] P. Buneman. A characterization of rigid circuit graphs. *Disc. Math.*, 9:205–212, 1974.

[19] L. Chen and Y. Yesha. Parallel recognition of the consecutive 1's property with applications. *J. Algorithms*, 12(3):375–392, 1991.

[20] L. Chen and Y. Yesha. Efficient parallel algorithms for bipartite permutation graphs. *Networks*, 22:29–39, 1993.

[21] D. Corneil and Y. Perl. Clustering and domination in perfect graphs. *Disc. Appl. Math.*, 9:27–39, 1984.

[22] D.G. Corneil, H. Lerchs, and L. Stewart. Complement reducible graphs. *Disc. Appl. Math.*, 3:163–174, 1981.

[23] E. Dekel and S. Sahni. Parallel scheduling algorithms. *Operations Research*, 31(1):24–49, 1983.

[24] E. Dekel and S. Sahni. A parallel matching algorithm for convex bipartite graphs and applications to scheduling. *J. Parallel Distributed Computing*, 1:185–205, 1984.

[25] M. Delattre and P. Hansen. Bicriterion cluster analysis. *IEEE trans. Patt. Analysis & Mach. Intel.*, PAMI2(4):277–291, 1980.

[26] J.S. Deogun, D. Kratsch, and G. Steiner. Graph clustering. Under preparation.

[27] E.S. Elmallah and C.J. Colbourn. Reliability of Δ-Y networks. In *Proc. Southeastern Conf. on Combinatorics, Graph Theory and Computing*, volume 48, pages 49–54, 1985.

[28] E.S. Elmallah and C.J. Colbourn. On two dual classes of planar graphs. *Disc. Math.*, 80:21–40, 1990.

[29] M. Farber. Characterization of strongly chordal graphs. *Disc. Math.*, 43:173–189, 1983.

[30] M. Farber. Domination, independent domination, and duality in strongly chordal graphs. *Disc. Appl. Math.*, 7:115–130, 1984.

[31] A. Farley, S. Hedetniemi, and A. Proskurowski. Partitioning trees: Matching, domination, and maximum diameter. *Int. J. Comp. Inf. Sci.*, 10(1):55–61, 1981.

[32] D.R. Fulkerson and O.A. Gross. Incidence matrices and interval graphs. *Pacific J. Math.*, 15:835–855, 1965.

[33] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W.H. Freeman and Company, 1979.

[34] F. Gavril. Algorithms for minimum coloring, maximum clique, minimum covering by cliques and maximum independent set of a chordal graph. *SIAM J. Comput.*, 1(2):180–187, 1972.

[35] F. Gavril. The intersection graphs of subtrees in trees are exactly chordal graphs. *J. Comb. Theory Ser. B*, 16:47–56, 1974.

[36] A. Gibbons and W. Rytter. *Efficient Parallel Algorithms.* Cambridge University Press, 1988.

[37] P.C. Gilmore and A.J. Hoffman. A characterization of comparability graphs and of interval graphs. *Canad. J. Math.*, 16:539–548, 1964.

[38] F. Glover. Maximum matching in a convex bipartite graph. *Naval Res. Logist. Quart.*, 14:313–316, 1967.

[39] M.C. Golumbic. The complexity of comparability graph recognition and coloring. *Computing*, 18:199–203, 1977.

[40] M.C. Golumbic. *Algorithmic Graph Theory and Perfect Graphs.* Academic Press, 1980.

[41] M.C. Golumbic. Algorithmic aspects of intersection graphs and representation hypergraphs. *Graphs and Combinatorics*, 4:307–321, 1988.

[42] M.C. Golumbic, D. Rotem, and J. Urrutia. Comparability graphs and intersection graphs. *Disc. Math.*, 43:37–46, 1983.

[43] C.C. Gotlieb and S. Kumar. Semantic clustering of index terms. *J. Assoc. Comput. Mach.*, 15(4):493–513, 1968.

[44] R. Greenlaw, H.J. Hoover, and W.L. Ruzzo. *Limits to Parallel Computation: P-completeness Theory.* Oxford University Press, 1995.

[45] M. Grötschel, L. Lovász, and A. Schrijver. Polynomial algorithms for perfect graphs. *Annals Disc. Math.*, 21:325–356, 1984.

[46] U.I. Gupta, D.T. Lee, and J. Y.-T. Leung. Efficient algorithms for interval graphs and circular-arc graphs. *Networks*, 12:459–467, 1982.

[47] P. Hansen and M. Delattre. Complete-link cluster analysis by graph coloring. *J. Amer. Statist. Ass.*, 73:397–403, 1978.

[48] P. Hansen and B. Jaumard. Minimum sum of diameters clustering. *J. of Classification*, pages 215–226, 1987.

[49] P. Hansen and B. Jaumard. Maximum-sum of splits clustering. Technical Report 88-07, University of Montreal, 1988.

[50] R. Horaud and T. Skordas. Stereo correspondence through feature grouping and maximal cliques. *IEEE trans. Pattern Anal. Machine Intell.*, 11(11):1168–1180, 1989.

[51] R.E. Jensen. A dynamic programming algorithm for cluster analysis. *Operations Research*, 17:1034–1057, 1969.

[52] R.M. Karp, E. Upfal, and A. Wigderson. Constructing a perfect matching is in random NC. *Combinatorica*, 6(1):35–48, 1986.

[53] R.M. Karp, E. Upfal, and A. Wigderson. The complexity of parallel search. *J. Comp. and Sys. Sciences*, 36:225–253, 1988.

[54] J.M. Keil and T.B. Brecht. Clustering in planar graphs. Technical Report 88-05, University of Saskatchewan, 1988.

[55] J.M. Keil and T.B. Brecht. The complexity of clustering in planar graphs. *J. Comb. Math. and Comb. Comp.*, 9:155–159, 1991.

[56] P. N. Klein. Efficient parallel algorithms for chordal graphs. In *Proc. 29th Annual IEEE Symp. Foundations of Computer Science*, pages 150–161, 1988.

[57] W.L.G. Koontz, P.M. Narendra, and K. Fukunaga. A branch and bound clustering algorithm. *IEEE trans. on Computers*, c-24(9):908–914, 1975.

[58] D.W. Matula. Graph-theoretic techniques for cluster analysis algorithms. In J.V. Ryzin, editor, *Classification and Clustering*, pages 95–129. Academic Press, 1977.

[59] D.W. Matula and L.L. Beck. Smallest-last ordering and clustering and graph coloring algorithms. *J. ACM*, 30(3):417–427, 1983.

[60] A. Moitra and R. Johnson. Parallel algorithms for maximum matching and other problems on interval graphs. Technical Report TR 927-88, Cornell University, 1988.

[61] C. Monma and S. Suri. Partitioning points and graphs to minimize the maximum or the sum of diameters. In *Proc. Sixth Int. Conf. Theory and Appl. of Graphs*, pages 899–912, 1989.

[62] S. Olariu, J.L. Schwing, and J. Zhang. Optimal parallel algorithms for problems modeled by a family of intervals. *IEEE trans. on Parallel and Distributed Systems*, 3(3):364–374, 1992.

[63] Y. Perl and S.R. Schach. Max-min tree partitioning. *J. ACM*, 28(1):5–15, 1981.

[64] Y. Perl and Y. Shiloach. Efficient optimization of monotonic functions on trees. *SIAM J. Alg. Disc. Meth.*, 4(4):512–516, 1983.

[65] Y. Perl and M. Snir. Circuit partitioning with size and connection constraints. *Networks*, 13:365–375, 1983.

[66] G. Ramalingam and C. Pandu Rangan. New sequential and parallel algorithms for interval graphs. *Infor. Proc. Letters*, 34:215–219, 1990.

[67] M.R. Rao. Cluster analysis and mathematical programming. *J. Amer. Stat. Ass.*, 66:622–626, 1971.

[68] F.S. Roberts. *Graph Theory and Its Applications to Problems of Society.* SIAM, 1978.

[69] D. J. Rose. Triangulated graphs and the elimination process. *J. Math. Anal. Appl.*, 32:597–609, 1970.

[70] R. Schrader. Approximations to clustering and subgraph problems on trees. *Disc. Appl. Math.*, 6:301–309, 1983.

[71] J. Spinrad, A. Brandstädt, and L. Stewart. Bipartite permutation graphs. *Disc. Appl. Math.*, 18:279–292, 1987.

[72] K. Takamizawa, T. Nishizeki, and N. Saito. Linear-time computability of combinatorial problems on series-parallel graphs. *J. ACM*, 29:623–641, 1982.

[73] A. Tucker. A structure theorem for the consecutive 1's property. *J. Comb. Theory Ser. B*, 12:153–162, 1972.

[74] J. Wald and C.J. Colbourn. Steiner trees, partial 2-trees, and minimum IFI networks. *Networks*, 13:159–167, 1983.

[75] W.J. Welch. Algorithmic complexity : Three NP-hard problems in computational statistics. *J. Stat. Computation & Simulation*, 15:17–25, 1982.

[76] Z. Wu and R. Leahy. An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation. *IEEE trans. Pattern Anal. Machine Intell.*, 15(11):1101–1113, 1993.

Appendix A: Graph Theory Terminology

Let $G = (V, E)$ be a simple unweighted undirected graph.

Let v_i and v_j in V, v_i is *adjacent* to v_j if $(v_i, v_j) \in E$.

For each vertex v in V let $N(v)$ be the *(open) neighborhood set* of v, $N(v) = \{u : (u, v) \in E\}$.

The *complement* of $G = (V, E)$ is $G^c = (V, E^c)$ where $E^c = \{(v_i, v_j) : (v_i, v_j) \notin E\}$.

$G_1 = (V_1, E_1)$ is a *(partial) subgraph* of $G = (V, E)$ if $V_1 \subseteq V$ and $E_1 \subseteq E$. If furthermore $\forall v_i$ and v_j in V_1: if $(v_i, v_j) \in E$ then $(v_i, v_j) \in E_1$, then G_1 is an *induced subgraph*. Unless otherwise specified we deal with induced subgraphs.

A *p-bond* is a maximal connected subgraph G_1 where every vertex has degree at least p in G_1.

A *p-component* is a maximal induced subgraph G_1 such that for any bipartition of G_1 at least p edges have ends in each partition.

A *k-clique* is a graph on k vertices with all possible edges.

A *k-independent set* is a graph on k vertices with no edges.

A *path* $P(v_i, v_j)$ is a sequence of vertices $\{v_i, u_1, u_2, ...u_p, v_j\}$ such that $(v_i, u_1) \in E$, $(u_l, u_{l+1}) \in E, 1 \leq l < p$, and $(u_p, v_j) \in E$. The *length of the path* is $p + 1$.

The *distance* between vertices v_i v_j, $dist(v_i, v_j)$, is the length of the shortest path between them.

The *diameter* of a graph G, $diam(G)$, is defined as $max_{v_i,v_j \in V} dist(v_i, v_j)$.

A graph G is *connected* if there is a path between any pair of vertices in it. Otherwise it is *disconnected*.

A *cycle* in G is a sequence of vertices $\{v_1, v_2, ...v_p\}$ such that $(v_i, v_{i+1}) \in E, \forall i, 1 \leq i < p$ and $(v_p, v_1) \in E$. A cycle is chordless if it has no more edges.

The *clique number* of G , $\omega(G)$, is the size of the largest clique in G.

The *chromatic number* of G, $\chi(G)$, is the fewest number of colors needed to color vertices of G such that no two adjacent vertices receive the same color.

The *stability number* of G, $\alpha(G)$, is the size of the largest independent set in G.

The *clique cover number* of G, $\kappa(G)$, is the size of the smallest possible clique cover of the vertices of G.

Appendix B: Graph Families

We deal mainly with perfect graphs in this thesis. In this appendix we define the perfect graph families we study.

For a given graph $G = (V, E)$, let G_A be the subgraph of G induced by the set of vertices $A \subseteq V$.

A graph $G = (V, E)$ is *perfect* if it satisfies any of the following properties [40]:
(P1) $\omega(G_A) = \chi(G_A)\ \forall A \subseteq V$.
(P2) $\alpha(G_A) = \kappa(G_A)\ \forall A \subseteq V$.
(P3) $\omega(G_A) * \alpha(G_A) \geq |A|\ \forall A \subseteq V$.
Thus if a graph is perfect its complement is perfect as well. Comparability and chordal graphs are perfect.

Comparability graphs are exactly the comparability graphs of posets, with an edge (x, y) if elements x and y are related. Equivalently, G is a comparability graph if there exists an orientation F of its edges such that for all $x, y, z \in V$, $xy \in F$ *and* $yz \in F \rightarrow xz \in F$ (see [40] and [41]). Bipartite and permutation graphs are comparability graphs.

The complements of comparability graphs, *cocomparability graph*, are intersection graphs of curves of continuous functions [42]. As well, G is a cocomparability graphs if there is an ordering of V such that for all $x, y, z \in V$, $x < y < z$ *and* $xz \in E \rightarrow xy \in E$ *or* $yz \in E$. Permutation and interval graphs are cocomparability graphs.

Permutation graphs are those graphs which are both comparability and cocomparability. Also a graph is a permutation graph if there is a pair P, Q of permutations of the vertex set such that there is an edge between vertices x and y if and only if the order of x and y is different in the two permutations.

A graph is *bipartite* if its vertex set can be partitioned into two independent sets X and Y.

A bipartite graph $G = (X, Y, E)$ is a *bipartite permutation* graph if there is an ordering of X which has the following two properties [71]:

> *The adjacency property:* for each vertex y in Y the neighborhood set of y in X consists of consecutive vertices.
>
> *The enclosure property:* for every pair u, v in Y such that $N(u)$ is a subset of $N(v)$, vertices in $N(v) - N(u)$ are consecutive.

If X satisfies the adjacency and enclosure properties, then so does Y.

Bipartite permutation graphs have another characterization also. Their vertices may be ordered so they have the so called *strong ordering* [71], which may be described as follows.

> A bipartite graph $G = (X, Y, E)$ has a *strong ordering* of its vertices if and only if its vertex set may be ordered to obey the following.

If (x_a, y_b) and (x_c, y_d) are edges in E, $x_a < x_c$, and $y_d < y_b$ then both (x_a, y_d) and (x_c, y_b) are edges in E.

If $G = (X, Y, E)$ is a bipartite graph and both X and Y have an ordering that satisfies the adjacency property then G is a *biconvex graph* ([16]).

Therefore bipartite permutation graphs are a proper subset of biconvex graphs.

Biconvex graphs are, in turn, a proper subset of *convex graphs* ([16] and [15]) which are bipartite graphs that have an ordering of X and Y such that the adjacency property is satisfied for one of X or Y.

Chordal Bipartite graphs contain convex graphs, and are those bipartite graphs that have no induced chordless even cycles ([40]).

Chordal graphs are the intersection graphs of subtrees of a tree (see [35] and [18]). They are also graphs with no induced cycle of length greater than three.

Strongly Chordal graphs [29] have a few characterizations. They may be characterized as those chordal graphs whose clique-vertex bipartite graph $G_B = (V, \mathcal{C}, \mathcal{E})$ is chordal bipartite. There is a vertex in V for every vertex in the chordal graph, and a vertex in $\mathcal{C}$ for every maximal clique in the chordal graph, and (v_i, C_j) is in $\mathcal{E}$ if and only if v_i is in the maximal clique C_j.

Undirected Path graphs are those chordal graphs that are the intersection graphs of paths in a tree.

Interval graphs are a subset of chordal graphs. They are the intersection graphs of intervals in a line. They are also precisely these graphs that are both chordal and cocomparability (see [37]).

If the lengths of the intervals are restricted to be equal then the intersection graph is a *unit interval graph.*

Split graphs are chordal graphs whose vertex set can be partitioned into a clique and an independent set.

Another family of chordal graphs is the family of *h-trees.* An *h-tree* is defined recursively as follows : an h-clique is an h-tree on h vertices, an h-tree on n vertices, $T(h, n)$, is constructed from $T(h, n-1)$ by making the new vertex adjacent to any h-clique in $T(h, n-1)$. Thus, trees are $1 - trees$.

Appendix C: Complexity Classes, Complete Problems, and Computation Models

In this appendix we give brief definitions of the complexity classes mentioned in this thesis, and describe the parallel computation model we use. More detailed explanations are found in [33] and [36].

For a certain model of *deterministic Turing machine* (DTM), let L_M be the language recognized by the DTM program M. Let $T_M(n)$ be the time complexity for M.

M is a *polynomial time program* if there is a polynomial p such that for all n in Z^+, $T_M(n) \leq p(n)$. The class $P = \{L : \exists$ *a polynomial time DTM program M for which* $L = L_M\}$.

A *decision problem* is one that has one of two solutions: *yes* or *no*. There is a correspondence between decision problems and languages, when decision problems are encoded reasonably. Thus we will describe a problem π as a *polynomial decision problem* to mean that there is an encoding for π and a DTM program M that recognizes the resulting language and has polynomial time complexity.

The class *NP* is defined formally as the class of all decision problems that can be solved by a polynomial time non-deterministic algorithm. Such an algorithm guesses a solution, then in polynomial time checks whether or not it is correct.

A decision problem π_1 is *polynomially reducible* to another decision problem π_2 (written $\pi_1 \alpha \pi_2$), whenever there is a function f from the set of all instances of π_1 to the set of all instances of π_2 that satisfies: f is computable by a polynomial time algorithm, and for all instances I of π_1, I has a *yes* answer if and only if $f(I)$ has a *yes* answer.

A problem, π, is said to be *NP*-complete if it is in *NP* and for all other decision problems π' in *NP* $\pi' \alpha \pi$. However; it is sufficient to prove that π is in *NP* and that there is an *NP*-complete problem π' that reduces to π.

The class *NC* is the class of decision problems for which there exists a parallel algorithm that takes $\mathcal{O}(\log^k n)$ parallel time, for a constant k, where n is the size of the problem, and uses a polynomial number of processors. Certainly if some problem π is not in P then it is not in *NC*. However some polynomial time problems do not readily admit parallelization. These problems form the class of *P-complete* problems. If an efficient parallel solution could be found for any P-complete problem then an efficient parallel solution exists for all *P*-complete problems. For search problems the analogous class is the class *FNC* [44].

The model of parallel computation we use is the *Concurrent Read Exclusive Write- Parallel Random Access Machine* model (*CREW-PRAM*). In the *PRAM* model all possible links between processors and memory locations exist. Although this model is more idealistic than realistic, simulating such a model on any practical computer costs only polylogarithmic time [36]. Processors work synchronously to execute the same program and communicate through a shared random access memory. Each processor is a random access machine. The PRAM model is also called a *single instruction multiple data stream* model $SIMD$. We use a variation of this model where concurrent reads to the same memory location are allowed but concurrent writes are disallowed.