uOttawa

L'Université canadienne
Canada's university

FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTOCTORALES

uOttawa

L'Université canadienne
Canada's university

FACULTY OF GRADUATE AND POSDOCTORAL STUDIES

Mustafa Hamid Alhashem

AUTEUR DE LA THÈSE / AUTHOR OF THESIS

Master of Computer Science

GRADE / DEGREE

School of Information Technology and Engineering

FACULTÉ, ÉCOLE, DÉPARTEMENT / FACULTY, SCHOOL, DEPARTMENT

The Book Embedding of Ordered Sets

TITRE DE LA THÈSE / TITLE OF THESIS

N. Zaguia

DIRECTEUR (DIRECTRICE) DE LA THÈSE / THESIS SUPERVISOR

CO-DIRECTEUR (CO-DIRECTRICE) DE LA THÈSE / THESIS CO-SUPERVISOR

EXAMINATEURS (EXAMINATRICES) DE LA THÈSE / THESIS EXAMINERS

G-V. Jourdan

N. Santoro

Gary W. Slater

Le Doyen de la Faculté des études supérieures et postdoctorales / Dean of the Faculty of Graduate and Postdoctoral Studies

THE BOOK EMBEDDING OF ORDERED SETS

Mustafa Alhashem

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
in partial fulfillment of the requirements for the degree of
Master of Computer Science *

Computer Science Department
School of Information Technology and Engineering
University of Ottawa

* The program is a joint program with Carleton University,
administrated by the Ottawa-Carleton Institute of Computer Science

Library and Archives Canada

Published Heritage Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et Archives Canada

Direction du Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 0-494-11200-X
Our file Notre référence
ISBN: 0-494-11200-X

TO MY PARENTS,

MY WIFE,

AND MY CHILDREN

AHMED AND ALI

Contents

List of Figures

List of Tables

List of Algorithms

Abstract

A large number of important problems in different areas can be expressed as graph layout problems whose objective is to determine a linear layout in such way that a certain objective cost is optimized. In a book embedding for an *ordered set*, the elements are embedded on the spine of the book to form a linear extension. The *pagenumber* is the minimum number of pages needed to draw the edges as simple curves such that edges drawn in the same page do not intersect. The book (stack) layout can be applied to several areas of computer science and engineering disciplines including ordered sets, interconnection networks, fault tolerant VLSI design, circuit designs, sorting permutations, complexity theory, and graph drawing.

In this thesis we investigate the book (stack) layout problem of ordered sets. The pagenumber problem is known to be NP-complete, even if the order of the elements on the spine is fixed. As a result, the pagenumber problem has been studied only for some restricted classes of ordered sets. A literature review on the pagenumber problem for graphs and ordered sets is presented. We also provide the first efficient algorithm for drawing the bipartite interval orders in the minimum number of pages needed. We also give an upper bound for the pagenumber of bipartite ordered sets and the pagenumber of the complete multipartite ordered sets with length 4 and 5.

Acknowledgements

I would like to express my gratitude to all those who gave me the possibility to complete this thesis.

I am deeply indebted to my supervisor **Dr. Nejib Zaguia** whose help, stimulating suggestions and encouragement helped me in my research and in writing of this thesis. His wide knowledge and his logical way of thinking have been of great value for me. His understanding, encouraging and personal guidance have provided a good basis for my thesis.

Special thanks are due to **Dr. Guy-Vincent Jourdan** and **Mr. Liva Rakotomalala** for many time spent with me discussing aspects of this work. They often offered me useful explanations and clarifications. I would like also to thank my friends namely, **Yednek**, **Thierry**, **Hery**, **Saied**, **Tanya** and **Raid** for being around me and cheering me up and who read part of my thesis and offered me some corrections.

I would like to thank **Saudi Aramco Company** and its affiliate **Aramco Services Company** for the financial support during the study. Special thanks are due to my manager **Dr. Ahmed Metwalli** and Staff Development Group Leader **Mr. Dennis Sundman** for nominating me for the master's degree program. Special thanks also go to my advisors, **Mr. Brumfield, Ms. Garden** and **Dr. Williams,** and their assistants **Ms. Johnson**, **Ms. Marker,** and **Ms. Kennedy**, respectively, for their support.

Finally, I would like to give my special thanks to my wife whose patient love enabled me to complete this work.

Chapter 1

Introduction

1.1 Background

In today's world, data processing has become one of the cornerstones of information technology. As more data is stored, simulated, and analyzed, the importance of the visualization of complex data has become very evident. The data sets which result from all this data storage, simulation and analysis are becoming increasingly difficult to interpret without appropriate visualization techniques and tools. *Visualization* [GEC98] is the process of transforming data, information, and knowledge into a visual form which makes use of the natural visual capabilities of humans. With the help of appropriate visualization techniques and tools, we can interact with large volumes of data quickly and effectively making it possible to discover hidden characteristics, patterns, trends, and valuable knowledge. Visualization has fundamentally changed the way we present and understand large, complex data sets. As a result of this widespread and fundamental impact of visualization, we now can better analyze such data sets and arrive at better informed decisions.

Complex data can appear in different forms and models. Deep understanding of the data type and their relations allows us to utilize the right techniques and tools to interpret them. One of many forms in which data may appear is that of the ordered set. An *ordered set* is a set of elements together with a set of relations on these elements. The relation of the ordered set satisfies the reflexivity, anti-symmetry and transitivity properties. Due to its natural properties, ordered sets play an increasing role in contemporary applications. This increase is well evidenced in many fields, such as computer science, the social sciences and operations research.

1.2 Visualization of Ordered Sets

The drawing of clear, comprehensible visualizations of data is of fundamental importance in all branches of science regardless of the exact type of the graph. This generates many interests in defining criteria for easily comprehensible and good drawings. Frequently used criteria, or aesthetics, for good drawings include the minimization of crossing between edges, the minimization of the area of the drawing and increasing symmetry. However, each type of drawing has different aesthetics requirements and it is often difficult to deal algorithmically with all aesthetics at the same time [BETT99]. As a result, most drawing algorithms are developed in such a way that certain aesthetics are dealt with separately.

Ordered sets can appear in many contexts in which the information to be viewed, processed and analyzed is actually a representation of ordered sets. Several drawing techniques exist for drawing ordered sets. The most common pictorial scheme consistently used to represent ordered sets is the *upward drawing*. It is a graph whose vertices correspond to elements and whose edges correspond to pairs of comparable elements. Using an upward drawing as starting point, we can construct the *directed covering graph*. This is done by suppressing its reflexivity, by way of disregarding the loop at each vertex and disregarding all nonessential edges (i.e. those implied by transitivity). It is possible to orient the directed covering graph in such a way that all arrows point upward and then erase all arrows. A line drawing will therefore be an upward drawing provided that it contains no horizontal edges, and no nonessential edges. Orders are drawn bottom-up: if an element x is smaller than y then there exists a path from x to y that is directed upwards (see Figure 1.1).

We can also draw ordered sets in the form of a book embedding. In the book embedding for an ordered set, the elements of the ordered set are embedded on the spine of the book to form a *linear extension* (A total ordering of the elements of an ordered set P that is consistent with the ordering of P). The *pagenumber* is the minimum number of pages needed to draw the edges as simple curves such that edges drawn in the same page do not intersect (see Figure 1.2).

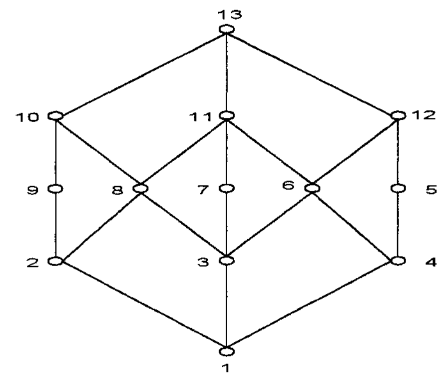


Figure 1.1: Upward drawing of an ordered set

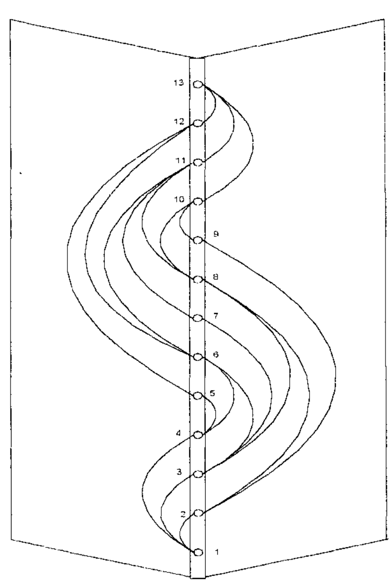


Figure 1.2: Book embedding of an ordered set

Book embedding was first defined for graphs. A *book embedding* of a graph consists of an embedding of its nodes along the spine of a book in a linear order and an embedding of its edges on pages so that edges embedded on the same page can be drawn as simple curves without intersections. Similar to a book embedding of an ordered set, the *pagenumber* of a graph is the minimum number of pages needed, taken over all permutations on the vertices of the graph. In the literature, the pagenumber is sometimes referred to as *stacknumber* or *thicknessnumber*.

1.3 Applications of the Pagenumber

A large number of relevant problems in different domains can be formulated as graph layout problems (see **Diaz et al.** [DPS02] for a survey). These include optimization of networks for parallel computer architectures, VLSI circuit design, information retrieval, numerical analysis, computational biology, graph theory and scheduling. Layout problems are a particular class of combinatorial optimization problems whose objective is to determine a linear layout of an input graph in such way that a certain objective cost is optimized.

The book layout (pagenumber) is being investigated on different types of graphs such as ordered sets [A96, AR96, H93, HP97, NP89, S89], directed graphs [GDLW02, HP99, HPT99], VLSI design [CLR87] and interconnection networks [KRSZ02]. There are several objectives for the pagenumber problem including saving resources in science and engineering fields. A solution to the pagenumber problem also aims to simplify the complex structures of scientific and engineering designs to make them more understandable. There are many contexts in which book embeddings for graphs occur.

1. Ordered Sets

For some problems, the underlying structure may be an ordered set [NP89]. For example, the vertices of the ordered set may correspond to steps in a computation and the edges directed into a vertex may correspond to necessary inputs for the computation to be carried out. As an example, we can consider the case where the computation is done on a computer with a single processor. In this example, each page in the book layout would represent a stack. The edges indicate the order in which the partial results are pushed and popped from the stacks. The pagenumber is the smallest number of stacks (pages) required to store the data in order for the computation to be executed.

In some applications, such as sorting using stacks and queues in parallel [CLR87], the set of feasible ordering of vertices on the spine is restricted to a certain subfamily of permutations. The book embedding problem of general graphs reduces to that for Hasse diagram (covering graphs of ordered sets) when the family coincides with the set of all linear extensions of a partially ordered set.

2. Sorting with Parallel Stacks

The pagenumber problem has additional applications to realizing fixed permutation of $\{1, ..., n\}$ with non communicating stacks in parallel [CLR8, EI71, KRSZ02]. The members are pushed onto any available stack and then removed to form the permutation in the order of 1 to n. Formally, we consider a bipartite graph G_n with vertices $\{x_1, ...x_n, y_1, ..., y_n\}$ and edges between each x_i and y_i. The problem of realizing the permutation π on $\{1, ..., n\}$ with k parallel stacks corresponds to embedding G_n in a k-page book, with its vertices embedded in the order of $x_1, ...x_n, y_{\pi(1)}, ..., y_{\pi(n)}$.

3. Fault Tolerant VLSI Design

The DIOGENES approach was proposed by **Rosenberg** [R83b] for the design of fault-tolerant VLSI processor arrays. It is one of the classical applications of book embedding [M94a]. The DIOGENES approach takes a prearranged interconnection network and implements it on a computer chip in a form of a book embedding. Wires are grouped into bundles that correspond to pages and vertices correspond to the set of fault-free processors on the chip. Even though the interconnection network may contain faulty processors, the chip has enough built-in redundancy to compensate for the expected number of processor failures. The faulty processors are dropped, and the fault-free ones are interconnected through the bundles. The bundles work as queues and/or stacks. If the bundles work as stacks, then the realization of an interconnection network needs a book-embedding of the interconnection network. A book embedding of the arranged network that uses few pages corresponds to more a hardware-efficient layout.

4. Fault Tolerant Processors Arrays

The following application is also in VLSI design [CLR87, KRSZ02] used to simplify the hardware complexity. Processors are embedded in a line and connected by hardware stacks or queues that run above the line. The problem of implementing arrays with stacks in the DIOGENES approach can be translated into a problem of embedding graphs in books. With embedded switches, group of wires run parallel to the line of processors such that each group of wires can implement a hardware stack of connections among processors. Each connection can only happen on one hardware stack. The processors are scanned from one side to the other. When a processor initiates a communication with another adjacent processor, the connection between them is pushed into one of the stack clusters. The connection is then popped from the stack while other connections that are currently in this cluster are shifted up one place. The problem is to recognize the desired interconnection graph using the minimum number of stack clusters. Each page of the book embedding corresponds to a hardware stack in the implementation and the condition that edges do not cross in the same page corresponds to the property of stacks.

5. Single-Row Routing

The pagenumber approach may also be applied to routing in multilayer printed circuit boards (PCBs). The complexity of routing multilayer PCBs problem is reduced in the following manner [CLR87]. Circuit elements are arranged in a regular grid, with wiring channels separating rows and columns of elements. The circuit's net lists are broken down so that every net connects elements in either a single row or a single column. As a result, the PCB can be diverted by routing each of its rows and each of its columns independently. One variation of this scenario is to prevent a net from running from the top of a row around to its bottom or from changing layers en route. This variant corresponds to the problem of embedding small vertex-degree graphs in a book.

6. Electrical Engineering

Another application to the pagenumber is in the VLSI circuit design [MOM98]. In this application, a chip is represented as hypergraph consisting of nodes corresponding to macrocells and hyperedges corresponding to a net connecting the cells. A printed circuit board usually consists of superimposed layers. A designer is required to place the macrocells on printed circuit board according to several design rules. Since crossing causes undesirable signals, avoiding crossing is one of the design requirements. It is therefore required to find ways to address the wirecrossing of the graph representing the chip. Consequently, crossing-wires must be laid out in different layers. The representing graph is decomposed into planar subgraphs. Each subgraph is then completely embedded on one layer which is not used by the other planar subgraphs.

Book embeddings have additional applications in several areas of theoretical computer science and its applications including:

- Sorting permutations [EI71,T72],
- Complexity theory [GKS89],
- Compact graph encodings [MR01],
- Compact routing tables [GH99] and
- Graph drawing [BSWW99].

1.4 Other Layouts

There exist other layouts for graphs in general and ordered sets in particular that have been investigated in the literature. The dual concept of the stack (page) layout is the queue layout. A *queue layout* consists of a total order of the vertices and a partition of the edges into queues, such that no two edges in the same queue are nested. The minimum number of queues in a queue layout of the graph is its *queue number*.

The queue layout of graphs [DW04, EI71], directed graphs [HP97, HP99, HPT99] and ordered sets [HP97, HP99, HPT99] have been extensively studied and investigated. Applications of the queue layout appear in many areas including sorting permutations [EI71, T72], parallel process scheduling [BCLR96], matrix computation [P92] and graph drawing [GM04].

However, we will only discuss the stack (book) layout in this thesis.

1.5 Motivation

Like many combinatorial optimization problems in computer science, the pagenumber problem of ordered sets is difficult and it is determined to be NP-complete even if the order of the nodes on the spine is fixed. Very few results on the pagenumber of ordered sets are known. The pagenumber of ordered sets is only known for a very small restricted set of classes, such as forest and series parallel planar ordered sets.

Extensive research and significant progress have been made in the visualization and the drawing of graphs and lattices. However, insufficient research and little progress have been made in the visualization and the drawing of ordered sets in a transparent manner despite its natural representations. Motivated by our interest in promoting abstract concepts such as ordered sets, we investigate the pagenumber of some new special classes, including interval orders and complete multipartite ordered sets.

1.6 Contributions

The main contributions of this thesis are the following:

1. We present surveys of the current state of the art with respect to the book layout problem for graphs (see chapter 3) and ordered sets (see chapter 4).
2. We investigate the pagenumber of interval orders (see chapter 5). Other research has been done for different restricted classes of ordered sets. We provide a polynomial algorithm for drawing bipartite interval orders in the minimum number of pages needed.
3. We also provide an upper bound for the pagenumber of bipartite ordered sets and the pagenumber of complete multipartite ordered sets with length 4 and 5 (see chapter 5).

1.7 Thesis Outline

The remainder of this thesis is organized as follows. Each chapter begins with an overview section which serves as an extended abstract for the chapter. Chapter 2 is dedicated as an introduction to the theory of ordered sets where several necessary terms and parameters of ordered sets are defined. We also introduce the class of interval orders, a focal point in this thesis.

In chapter 3, we provide a review of recent literature on the pagenumber of graphs. We present the known results to the pagenumber of graphs. Furthermore, a review of the complexity of the pagenumber of graphs along with some bounds on the pagenumber of certain families of graphs is presented.

Chapter 4 provides a literature review on the pagenumber of ordered sets. In addition to the cases where computing the pagenumber is known, we provide some lower and upper bounds, as well as a variety of examples of ordered sets widely discussed in the literature of the pagenumber.

In chapter 5, we discuss the main contributions of the thesis. We present an efficient algorithm that computes the pagenumber of bipartite interval orders. We also give an upper bound for the pagenumber of bipartite ordered sets and for the pagenumber of complete multipartite ordered set with length 4 and 5.

In chapter 6, we discuss some new research directions and open problems on the pagenumber of ordered sets. We discuss the effect of adding and deleting elements from ordered sets on the pagenumber. We also discuss the need for transformation techniques between optimal linear extensions. Finally, we discuss the idea of page critical by removing vertices or edges from the ordered sets.

In this thesis, we only consider finite ordered sets.

Chapter 2

Basics on Ordered Sets

2.1 Overview

This chapter provides an introduction to ordered sets. We cover the basic definitions about ordered sets, its parameters and its invariants. In addition, we give some basic concepts about drawing ordered sets. We also introduce some classes of ordered sets including lattices, series parallel ordered sets, N-free ordered sets, interval orders and semi-order. We finally define the pagenumber of graphs and ordered sets.

2.2 Definitions

A set with a *partial order* on it is called a *partially ordered set*, an *ordered set*, or, often, simply a *poset*. A *partial order* on set P is a binary relation $\leq$ on P such that the following conditions are satisfied for all x, y and z in P:

a) Reflexivity: $x \leq x$.
b) Anti symmetry: If $x \leq y$ and $y \leq x$, then $x = y$.
c) Transitivity: If $x \leq y$ and $y \leq z$, then $x \leq z$.

Examples of ordered sets include the integers and real numbers with their ordinary ordering, subsets of a given set ordered by inclusion, strings ordered lexicographically, and natural numbers ordered by divisibility.

For $x \neq y$ in the ordered set P, we say x is *comparable* to y if either $x < y$ or $x > y$. Otherwise, x is *noncomparable* to y and we write $x \parallel y$.

A *chain* in an ordered set P is a subset C of P for which any two elements of C are comparable. An *antichain* in an ordered set P is a subset A of P such that any two distinct elements of A are noncomparable. In Figure 2.1, $\{a, b, e, g, i\}$, $\{a, c, e, g, h\}$ and $\{a, d, f, h, i\}$ are chains in P since every element is comparable to the other elements. However, $\{b, c, d\}$, $\{e, f\}$ and $\{g, h\}$ are antichains in P since all elements are noncomparable to each others.

2.3 Drawing Ordered Sets

Traditionally, the elements of the ordered set P are represented by small circles and the covering relations are represented by interconnecting lines.

We write $x \prec y$ (or $y \succ x$) to indicate that x is covered by y (or y covers x) if $x < y$ and $x < z \leq y$ implies $z = y$. This means that there is no element z of P with condition $x < z < y$. We also call y an upper cover of x and x a lower cover of y. We also write (x, y) to denote an edge that represents a covering relation. The *covering graph* of P, denoted by $cov(P)$, is the graph whose vertices are the elements of P and the pair $\{x, y\}$ forms an edge in $cov(P)$ if $x \succ y$ or $y \prec x$.

Let P be an ordered set and let x be in P. The set of *successors* of x in P, denoted *Succ*(x), is the set of all elements y in P such that $x \leq y$. The set of *immediate* successors of x in P, denoted *ImmSucc*(x), is the set of all elements y in P such that x is covered by y.

The set of *predecessors* of x in P, denoted *Pred*(x), is the set of all elements y in P such that $x \leq y$. The set of *immediate* predecessors of x in P, denoted *ImmPred*(x), is the set of all elements y in P such that x is covered by y ($y \succ x$).

It is often convenient to specify ordered sets by means of drawn diagrams of the covering graph in the Euclidean plane. The *upward drawing*, *Hasse diagram* H(*P*), is *one* of the top common schemes for drawing an ordered set among the graphical data structures. The *upward drawing* is defined as a *cov*(*P*) drawn in the plane such that:

i. The element x is lower than the element y whenever $x \prec y$.

ii. The edge (x, y) is drawn straight and does not cross any other element of P.

Further, it is drawn so that it does not have loops and arcs implied by the transitivity and all arcs point upward eliminating arrowheads.

An ordered set is *planar* if it admits an upward drawing in which no edges cross. The *covering graph* of a planar ordered set is an *upward planar digraph*. Figure 2.1 shows an example of a planar upward drawing of an ordered set.

An element x is called a *minimal element* of P if x has no lower covers. The set of all minimal elements of P is denoted by *min*(*P*). Similarly, an element y is called a *maximal element* of P if y has no upper covers. The set of all maximal elements of P is denoted by *max*(*P*).

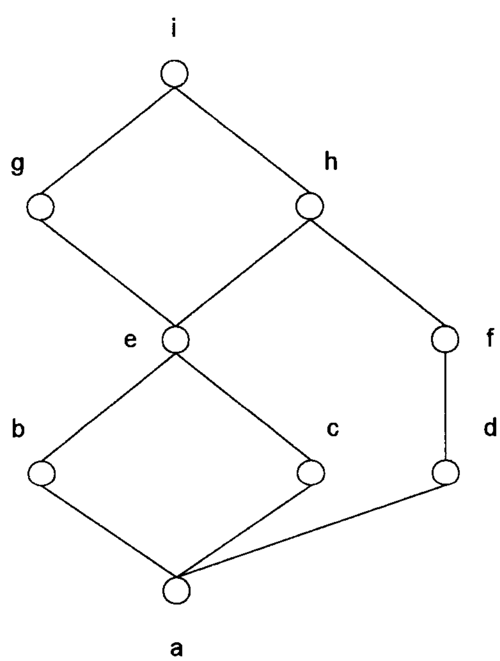


Figure 2.1: Planar upward drawing of an ordered set

Let $L_0 = min(P)$. For each integer $i > 0$, we define the i-th level of P *as*,

$$L_i = min\,(P - \bigcup_{j=0}^{i-1} \mathrm{Lj})$$

We define the *height*(P) as the size of the longest chain of P - 1. It is clear that there are exactly *height* (P) + 1 levels for P.

Similarly, we define *width*(P) as the size of largest antichain. In Figure 2.1, the longest chain has 5 elements while the longest antichain has 3 elements. Hence, *height*(P) = 4, *width*(P) = 3 and the ordered set has 5 levels.

2.4 Linear Extensions

A total ordering of the elements of an ordered set P is called a *linear extension* of P if it is consistent with the ordering of P, *i.e.* $x \leq y$ in L whenever $x \leq y$ in P and L is a chain. There are several ways to generate a linear extension of an ordered set. Algorithm 2.1 illustrates a way of how to generate a linear extension of an ordered set P.

Algorithm. *Linear Extension Generator*

```
/*      Input P: an ordered set with n elements      */
/*      Output L: a linear extension of P            */
        L = []
        for i = 1 to n do
            select(x_i ∈ min(P))
            L = L + x_i
            P = P \ {x_i}
```

Algorithm 2.1: Linear Extension Generator

A linear extension $L = \{x_1 < x_2 < \ldots < x_m\}$ of P is *greedy* if it is constructed inductively as follows: select x_{i+1} minimal in $P - \{x_1, \ldots, x_i\}$ that is comparable to x_i whenever possible. Algorithm 2.2 illustrates how to generate a greedy linear extension of an ordered set P. The following linear extensions L_1, L_3, L_7, L_8, L_{10} and L_{11}, shown in Figure 2.2, are greedy linear extensions.

Algorithm. Greedy *Linear Extension Generator*

* Input = P : an ordered set with n elements */

* Output = L a linear extension of P */

L = []

for i = 1 to n do

if $min(P) \cap next(x_{i-1} \neq \emptyset)$

select $(x_i \in min(P) \cap next(x_{i-1}))$

else

select$(x_i \in min(P))$

$L = L + x_i$

$P = P \setminus \{x_i\}$

Algorithm 2.2: Greedy Linear Extension Generator

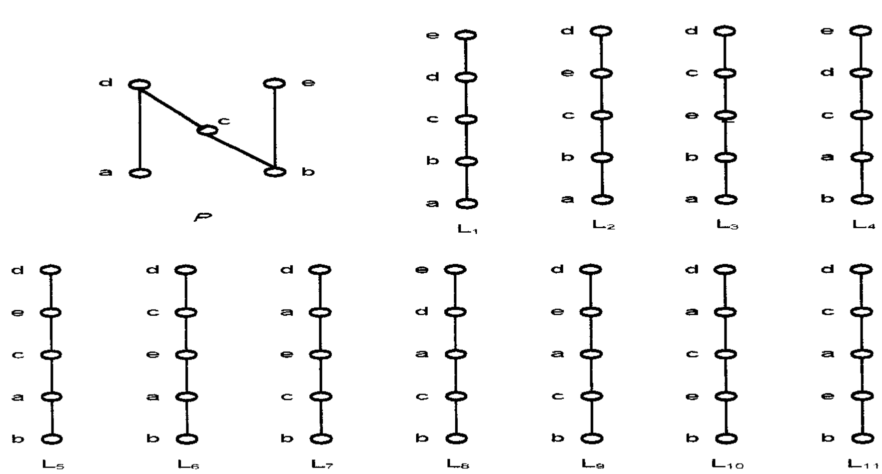


Figure 2.2: Linear extensions of an ordered set

2.5 Classes of Ordered Sets

There are several interesting classes of ordered sets. This section introduces some classes of ordered sets which are relevant to the subject of this thesis.

2.5.1 Lattices

The lattices class is an essential class of ordered sets. Lattices play an important role in various areas, for example representation theory, coding theory, geometry and algebraic number theory.

The *supremum* of a given set is the least element which is greater than or equal to each element of the set. Consequently, it is also referred to as the least upper bound. Dually, the *infimum* of a subset of some set is the greatest element that is smaller than all other elements of the subset. Consequently, the term greatest lower bound is also commonly used to refer to the infimum.

An ordered set *P* is a *lattice* if every finite subset *S* has supremum (join) denoted *sup*(*S*) and infimum (meet) denoted *inf*(*S*). This is equivalent to the existence of the supremum and the infimum for each pair of the elements of *P*. For instance, the ordered set in Figure 2.3(a) is not a lattice because *sup*({*2*, *3*}) does not exist. However, the ordered set in Figure 2.3(b) is a lattice.

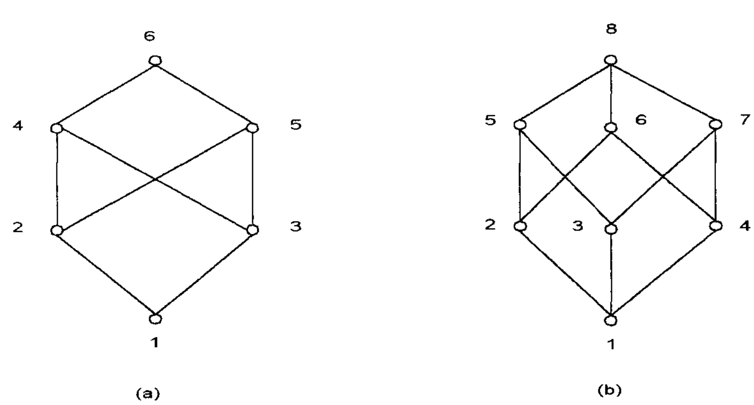


Figure 2.3: A non lattice ordered set (a) and a lattice (b)

2.5.2 Series Parallel Ordered Sets

The class of *series parallel* ordered sets is one class of ordered sets which generates a lot of interest due to its natural correspondence to many real applications in fields such as networking and circuit design. An ordered set P is *series parallel* if P can be constructed from singletons using only the operations of disjoint sum and linear sum.

The *linear sum* $P \oplus Q$ of the two disjoint ordered sets P, Q is an ordered set on $P \cup Q$, where $a \leq b$ if and only if

i. $a \leq b$ in P, or
ii. $a \leq b$ in Q, or
iii. $a \in P$ and $b \in Q$.

The *disjoint sum* $P \otimes Q$ of P, Q is defined the same way but with the elimination of the third condition of the linear sum. The ordered sets shown in Figure 2.1 and Figure 2.4 are series parallel ordered sets.

An important and interesting characterization of series parallel ordered sets is the *Series-Parallel-N theorem* [VTL82]: A finite ordered set is series-parallel if and only if it contains no subset isomorphic to N (see section 2.5.3 N-Free ordered sets).

2.5.3 N-Free Ordered Sets

The N-Free ordered sets class is another special class of ordered sets. We call an ordered set an *N-Free ordered set* if there is no subset $\{a, b, c, d\}$ whose only ordering relations are given by $a \prec c$, $a \prec d$ and $b \prec d$, and there are no further comparabilities between these elements.

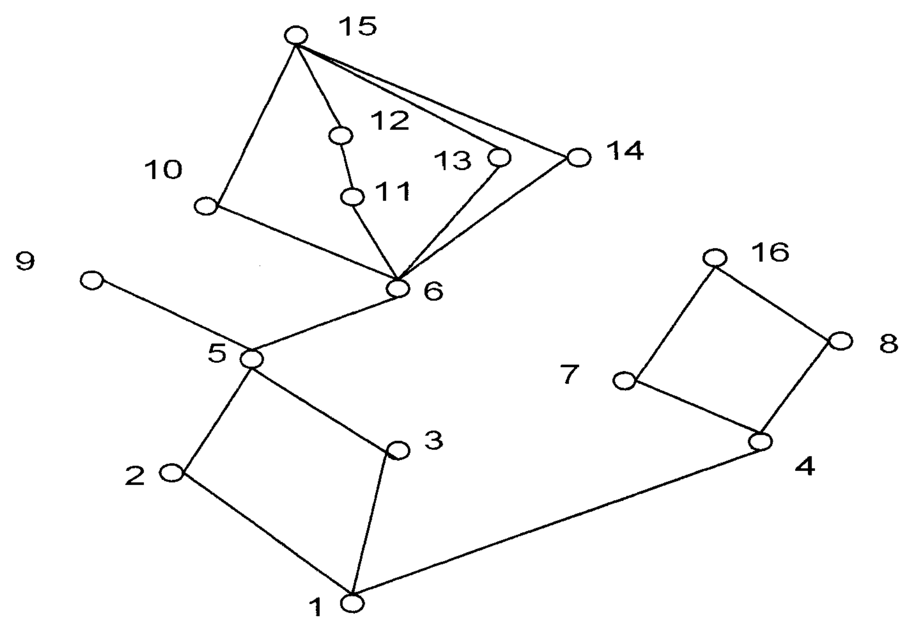


Figure 2.4: Series parallel planar ordered set

Clearly, a series parallel ordered set is N-Free. Figure 2.2 illustrates an N-Free ordered set which is not a series parallel ordered set. *N*-Free ordered sets generate a lot of interest due to the fact that removal of this structure (N relation) from the ordered sets results in improved and easier solutions for several problems.

N-Free ordered sets play historical role on the theory of ordered sets [R86]. For several combinatorial optimization problems, there are effective solutions for N-Free ordered sets.. Well known examples are the following:

- *Chain-Meet Antichain Theorem* [G83]. In a finite ordered set every maximal chain meets every maximal antichain if and only if it is N-Free.
- *Jump Number Theorem* [R83a]. For a finite N-Free ordered set, every greedy linear extension is jump-optimal and every jump optimal linear extension is greedy as well.

2.5.4 Interval Orders

The interval orders class is also a special class of ordered sets. An interval representation of an ordered set (P, <) is a function that assigns to each element u in P an interval on the real line I_u such that $u < v$ if and only if each point of I_u is less than every point in I_v. If an ordered set (P, <) has an interval representation then we call (P, <) an interval order. Figure 2.5 illustrates an example of an interval order with its interval representation. We can represent intervals by either lines or rectangles.

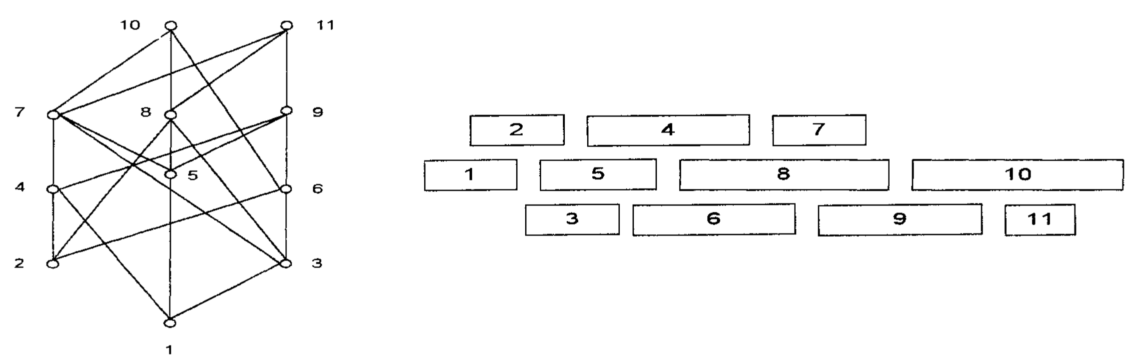


Figure 2.5: Interval representation

Interval orders have very nice characterizations which give more information about the structure and make them more understandable [F92]. An ordered set P is an interval order if and only if P does not contain a $2 \otimes 2$ as induced suborder, that is, a subset $\{u, v, x, y\}$ of P with $u < v$, $x < y$ are the only comparabilities among these elements (see Figure 2.6(a)). Therefore, an *interval order* is an ordered set (P, <) such that $w, x, y, z \in P$ with $w < x$ and $y < z$ imply either $w < z$ or $y < x$.

In any interval order P the following two important conditions also hold:

- The sets of predecessors are linearly ordered with respect to inclusion. That is, for all $x, y \in P$, *either* $Pred(x) \subseteq Pred(y)$ or $Pred(x) \supseteq Pred(y)$.
- The sets of successors are also linearly ordered with respect to inclusion. That is, for all $x, y \in P$, *either* $Succ(x) \subseteq Succ(y)$ or $Succ(x) \supseteq Succ(y)$.

The Semi orders class is a subset of the interval order class. For a *semi-order P* (**Felsner** [F92]):

- *P is* an interval order that does not contain a $1 \otimes 3$ as induced suborder, that is, a subset $\{u, v, w, x\}$ of P with $u < v < w$ and no more comparabilities (see Figure 2.6(b)).
- The *Pred-order* and the *Succ-order* of *P* are identical.

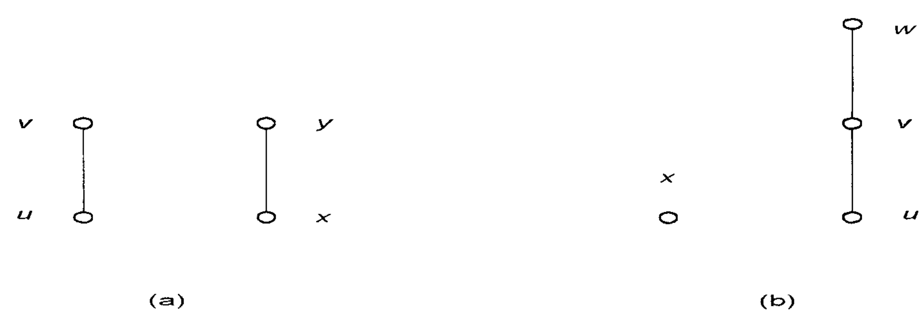


Figure 2.6: The partial orders $2 \otimes 2$ and $1 \otimes 3$

2.6 The Jump Number of Ordered Sets

Let the elements of *P* represent certain jobs to be performed one at a time by a single processor while the order of *P* imposes precedence constraints upon these jobs. A job performed immediately after one which is not constrained to precede it requires a *jump* or *setup* which involves some fixed additional cost [IOS96].

Formally, the *jump number* of an ordered set *P* is defined as follows. Let *L* be a linear extension of *P*,

$$jump\ (L, P) = |\ \{\ (a, b) : a \succ b(\mathrm{L}) \text{ and } a \parallel b\ (P)\ \}\ |$$

$$jump\ (P) = min\{\ jump(L, P) \text{: } L \text{ a linear extension of } P\ \}$$

An optimal linear extension of *P* is just a schedule of the jobs which minimizes the number of 'setups' between unrelated jobs. We call a linear extension a *jumpnumber* optimal if *jump*(*L*, *P*) = *jump*(*P*). The jumpnumber of the ordered set given in Figure 2.2

is two. Table 2.1 shows the jumpnumber of each linear extension of the ordered set given in Figure 2.2. Notice that all greedy linear extensions produce a jumpnumber which is optimal since it is an N-Free ordered set (see section 2.5.3).

Table 2.1: Jump number of linear extensions

L	**Jump#**	L	**Jump#**	L	**Jump#**
L_1	2	L_5	4	L_9	3
L_2	3	L_6	3	L_{10}	2
L_3	2	L_7	2	L_{11}	2
L_4	3	L_8	2	P	2

2.7 The Pagenumber

A *book embedding* of a graph G consists of an embedding of its nodes along the spine of a book and embeddings of its edges on pages so that edges embedded on the same page do not intersect. The *pagenumber* of G, *page(G)*, is the minimum number of pages needed, taken over all permutations on the vertices of G.

Equivalently, the vertices can be embedded on a circle with the edges being chords of the circle and the chords assigned to pages so that no two intersecting chords are assigned to the same page.

In a book embedding for an ordered set P, the vertices of P are embedded on the spine of the book to form a *linear extension* that respects the ordering relation. The pagenumber is the minimum number of pages needed.

Formally, the *pagenumber*, *page*(P), is defined as follows. Let £ (P) denote the set of all linear extensions of P. A book embedding of an ordered set P with respect to $L \in$ £ (P) is the embedding of the Hasse Diagram, H(P), with its vertices (i.e. elements of P) placed on the spine with respect to L. The pagenumber *page*(P, L) of P with respect to L is the smallest number k of pages such that H(P) has a book embedding on k pages. So,

$$Page(P) = min\{page(P, L) : L \in £\,(P)\}.$$

The *pagewidth* is the maximum number of edges that cross any line perpendicular to the spine of the book. An edge (x, y) is a spine edge if it is drawn on the spine; i.e. $x \prec y$ in L. We can draw a spine edge on any page of a book embedding of P.

The pagenumber problem is NP-complete even if the order of the nodes on the spine is fixed [A96].

Chapter 3

The Pagenumber of Graphs

3.1 Overview

This chapter surveys the pagenumber problem for graphs. We present the known results as well as the complexity of the pagenumber of graphs. In addition, we present some bounds on the pagenumber of certain families of graphs. It is worth mentioning:

- Some of these graphs are covering graphs and therefore give lower bounds for the corresponding ordered sets.
- A direct relation exists between ordered sets and graphs. For example, the covering graph of a planar ordered set is an upward planar digraph.

The necessary terms and definitions are introduced as needed in this chapter. Proofs are omitted and interested readers are referred to the literature.

3.2 Literature Review

The problem of computing the pagenumber of different types of graphs has been extensively investigated including computing the pagenumber of directed graphs [GDLW02, HP99, HPT99] and ordered sets [A96, AR96, H93, HP97, NP89, S89].

Bernhart and **Kainen** [BK79] were first to investigate the pagenumber of a graph which they called the book thickness. They showed that the pagenumber is equal to zero if and only if the graph is a path since its edges can be drawn on the spine. The book embedding problem for graphs is known to be difficult. **Garey et al.** [GJMP80] showed that even if a vertex ordering is given, it is NP-complete to find the optimal number of pages in which the edges can be embedded. Moreover, **Chung et al.** [CLR87] showed that it is NP-complete to determine if an arbitrary graph can be embedded in two pages.

3.2.1 Outerplanar Graphs

A graph is *outerplanar* if it can be embedded in the plane so that every vertex lies on the unbounded outside face of a circle and edges lie inside face such that edges do not intersect (see Figure 3.1). **Bernhart** and **Kainen** [BK79] showed that the 1-page graphs are exactly the outerplanar graphs. This can be demonstrated by cutting the circle between any two vertices and then opening it out to form a line such that it can be embedded in one page. Conversely, given a one-page embedding of G, we can pass a line through the vertices of G in their order in the embedding. We then join the ends of the line together to form a circle demonstrating G's outerplanarity.

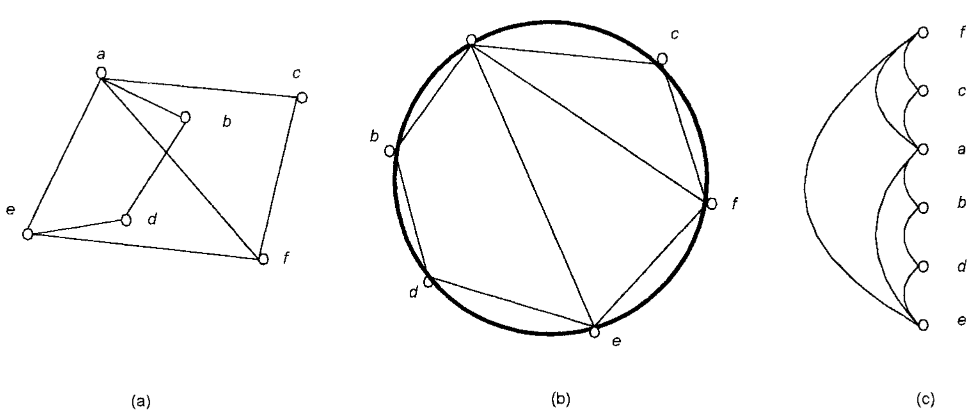


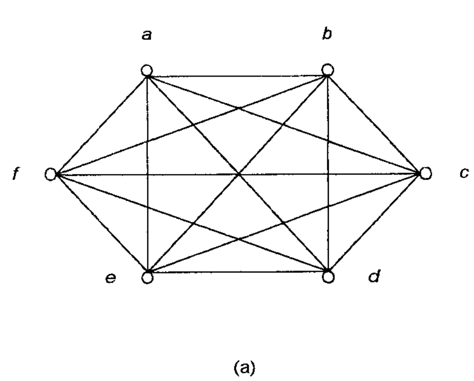
Figure 3.1: An outerplanar graph

3.2.2 Subhamiltonian Graphs

A *hamiltonian circuit* is a graph cycle that visits each node exactly once. A graph is *planar subhamiltonian* if it is a planar graph that can be edge-augmented to have a hamiltonian cycle, yet remain planar. A *subhamiltonian* graph is basically a subgraph of a hamiltonian graph. **Bernhart** and **Kainen** [BK79] showed that a graph G has pagenumber ≤ 2 if and only if it is a subgraph of a hamiltonian planar graph. The two-page book embedding is constructed as follows. Edges outside the circuit are drawn in one page, edges inside the circuit are drawn in a second page and edges on the circuit may be drawn in either page. A graph which is not planar requires no less than 3 pages.

3.2.3 Complete Graphs

A *complete graph* K_n on n vertices is a graph in which every pair of vertices is adjacent. **Bernhart** and **Kainen** [BK79] determined some of the properties of the pagenumber of complete graphs. **Chung et al.** [CLR87] showed that the complete graph K_n is embeddable in $\lfloor n/2 \rfloor$ pages, each page being of width at most n. Figure 3.2 illustrates a book embedding of the complete graph K_6 in three pages. In general, **Malitz** [M94a] proved that any graph with m edges has pagenumber $O(\sqrt{m})$.



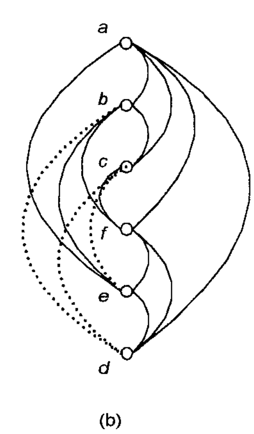


Figure 3.2: The Complete graph K_6

3.2.4 Planar Graphs

The maximum pagenumber of planar graphs was a subject of intense investigation. A *planar graph* is one that can be drawn on a plane in such a way that there are no edge crossings. **Bermhart** and **Kainen** [BK79] conjectured that the pagenumber of a planar graph can be arbitrarily large. **Buss** and **Shot** [BS84] disproved this conjecture by presenting an algorithm which embeds all planar graphs in 9 pages. The algorithm is based on Whitney's theorem which shows that every planar triangulation with no separating triangles has a hamiltonian cycle. **Heath** [H84] improved this result by providing an efficient algorithm with $O(n^2)$ complexity to embed all planar graphs in 7 pages. The algorithm uses a method of *peeling* of the graph into levels to reduce the pages to seven. **Istrail** [I86] managed to reduce the number to six pages.

Yannakakis [Y86,Y89] finally showed that any planar graph can be embedded on four pages in a linear time. This bound is tight since he showed that there are planar graphs that cannot be embedded in three pages. The algorithm was built on **Heath**'s approach by combining it with a different edge coloring method to reduce the number of pages to five. A different node laying out algorithm was used to reduce the number of pages to 4.

3.2.5 Series Parallel Graphs

A *series parallel graph* is a graph that can be obtained by a sequence of edge operations, series and parallel, from a graph consisting of only one edge.

- A *parallel operation* on an edge e replaces e by two parallel edges with the same endpoints as e.
- A *series operation* on an edge e replaces the edge by a path of length two.

Series parallel graphs are always planar. **Chung et al.** [CLR87] showed that the maximum pagenumber of a series parallel graph is two. **Chung et al.** [CLR87] also provided optimal or near-optimal embedding of a variety of families of graphs including trees, grids, X-tree, cycle shifters and permutation networks.

3.2.6 Depth-*n* Pinwheel Graphs *P*(*n*)

The depth-*n* pinwheel graph *P*(*n*) has $2n$ vertices

$\{x_1, x_2, \ldots, x_n\}$

and

$\{y_1, y_2, \ldots, y_n\}$

and edges connecting each pair of vertices of the form:

$x_i - y_i, \quad 1 \leq i \leq n$

$x_i - y_{n-i+1}, \quad 1 \leq i \leq n$

$x_i - x_{i+1}, \quad 1 \leq i < n$

$y_i - y_{i+1}, \quad 1 \leq i < n$

Chung et al. [CLR87] showed that the graph *P*(*n*) is 3-page embeddable in such a way that one page has width 2 and the other two each have width 4 (see Figure 3.3). Another 3-page embedding of *P*(*n*) is also possible with pagewidths 4, 3, and 1, respectively.

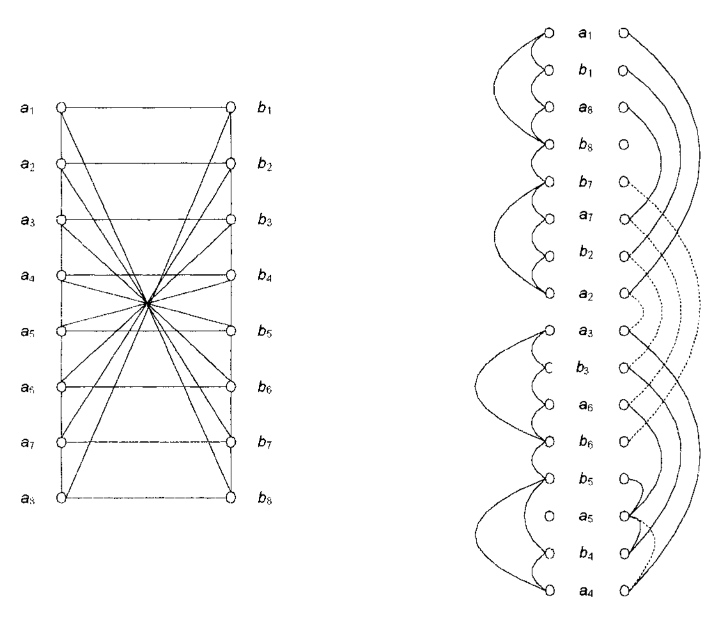


Figure 3.3: The Depth-8 pinwheel graph *P*(8)

3.2.7 Depth-d X-tree Graphs

The *depth-d X-Tree* X(*d*) is the edge augmentation of the depth-*d* complete binary tree that adds edges going across each level of the tree in left-to-right order. X-trees are planar and subhamiltonian. As a result, the depth-*d* X-Tree admits a 2-page embedding [CLR87], with one page of width 2*d* and one of width 3*d* (see Figure 3.4).

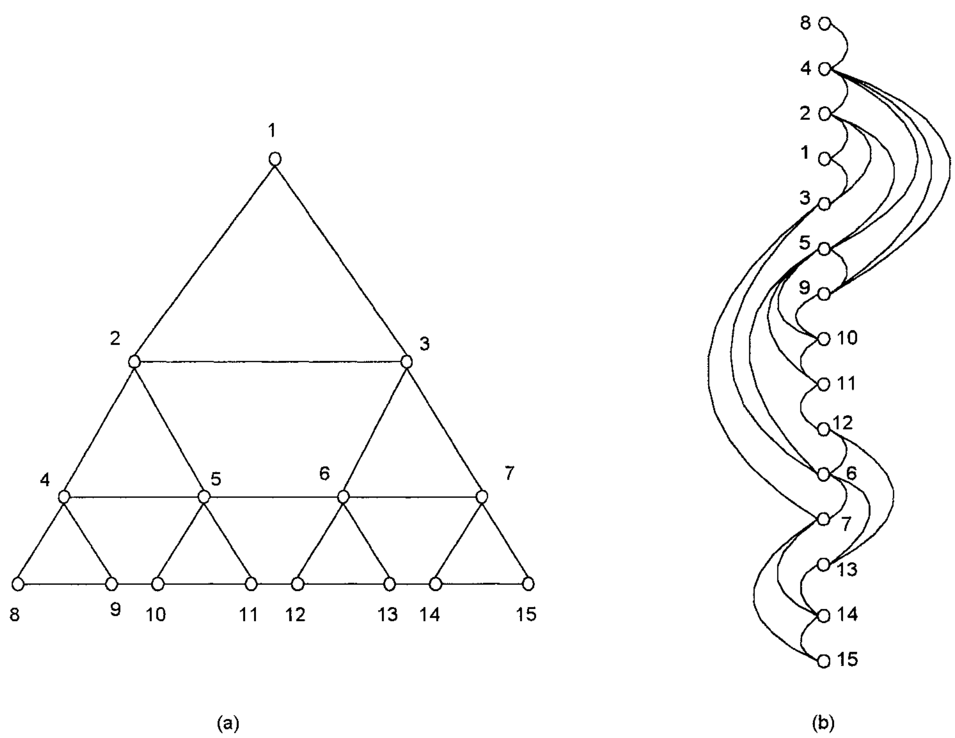


Figure 3.4: Depth-4 X-tree

3.2.8 Square Grids Graphs

Similarly, square grids are also planar and subhamiltonian. Consequently, $n \times n$ square grids admit 2-page embedding each with width n. The augmented hamiltonian cycle is formed by row-by-row alternating between left-to right and right-to-left (see Figure 3.5).

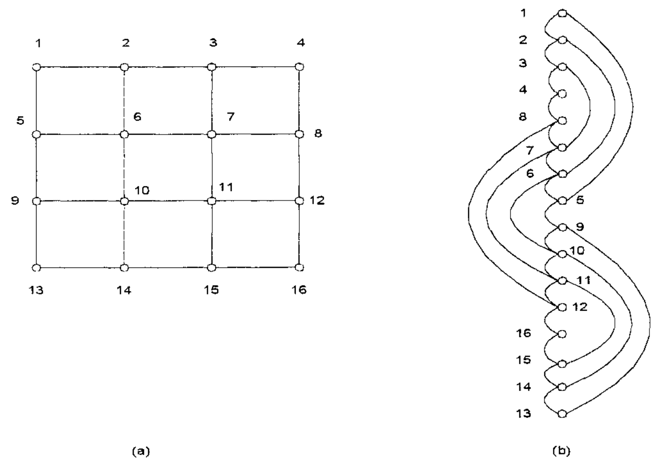


Figure 3.5: Square grids

3.2.9 Halin Graphs

A graph G is a *Halin graph* if the following are true:

- Every vertex in G has degree 3 or greater.
- G can be decomposed into a spanning tree T of G and a cycle C through the leaves of T.
- G has a planar embedding in which C forms the boundary of the infinite face.

Halin graphs are planar and subhamiltonian graphs. **Ganley** [G95] proved that if G is a Halin graph, then *page*(G) ≤ 2 and that there are Halin graphs that require 2 pages. For example, K_4 is a Halin graph that requires 2 pages.

3.2.10 Bipartite Graphs Planar

A *bipartite graph* G has two disjoint sets of vertices, V_1 and V_2, with the property that each edge in G has one vertex in V_1 and one vertex in V_2. **Fraysseix et al.** [FMP95] showed that the pagenumber of a bipartite planar graph is 2. They also showed that *page* (G) ≤ 2 for any planar graph with quadrilateral faces. A polygon (2-dimensional figure) with four sides has quadrilateral faces.

3.2.11 Complete Bipartite Graphs

A *complete bipartite graph* is a bipartite graph in which every vertex in V_1 is connected by an edge to every vertex in V_2. **Muder et al** [MWW88] gave an upper bound of $\lceil (m+2n)/4 \rceil$ for the pagenumber of the complete bipartite graph $K_{m,n}$ which was then improved upon by **Enomoto et al** [ENO97]. However, the pagenumber of complete bipartite graph has not yet been determined.

3.2.12 *K*-Trees Graphs

A k-tree is defined inductively in the following way:

- The k-vertex complete graph K_k is a k-tree.
- If G is a k-tree, then adding a new vertex to G that is adjacent to k vertices that induce a K_k in G results in a k-tree.

Ganley and **Heath** [GH01] showed that k-trees have pagenumber that is, at most, k+1 and that there are k-trees that require k pages. Figure 3.9 illustrates a 3-page embedding of the 2-tree graph.

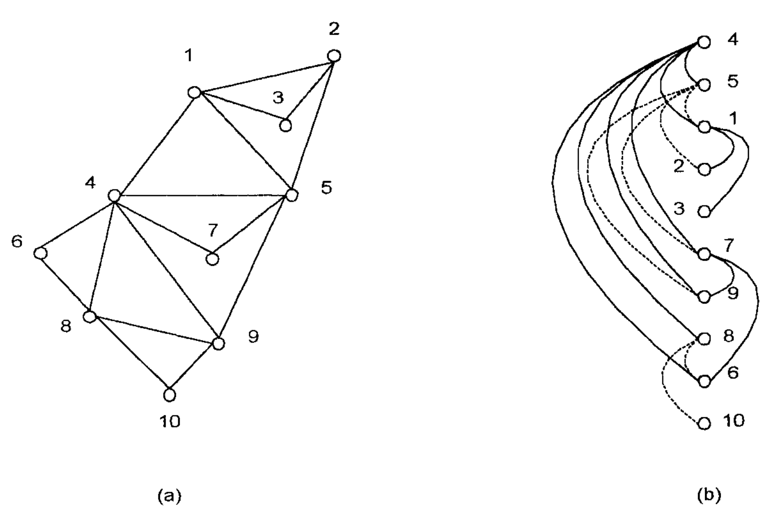


Figure 3.6: 2-Tree book embedding

3.2.13 Benes Permutation Networks

A Benes network graph is a subgraph of FFT network and Banyan network graphs. Let n be a power of 2. The *n-input Benes network* $B(n)$ is the graph defined inductively as follows.

- $B(2)$ is the complete bipartite graph $K_{2,2}$ on two input vertices $i_{1,1}$ and $i_{1,2}$ and two output vertices $o_{1,1}$ and $o_{1,2}$.
- $B(2)$ is obtained by taking two copies of $B(n/2)$ as well as n new input vertices, $i_{n,1}, i_{n,2}, \ldots, i_{n,n}$ and n new output vertices, $o_{n,1}, o_{n,2}, \ldots, o_{n,n}$. For each $1 \leq k \leq n$, one adds edges that create one copy of $K_{2,2}$ with "input" $i_{n,k}$ and $i_{n,k+n/2}$ and "output" $i_{n/2,k}$ and $i'_{n/2,k}$ and one copy of $K_{2,2}$ with "input" $o_{n/2,k}$ and $o'_{n/2,k}$ and "output" $o_{n,k}$ and $o_{n,k+n/2}$.

Benes networks are nonplanar graphs. Figure 3.7 depicts the 4-input Benes network. **Chung et al.** [CLR87] proposed that the Benes network $B(n)$ admits a 3-page book embedding.

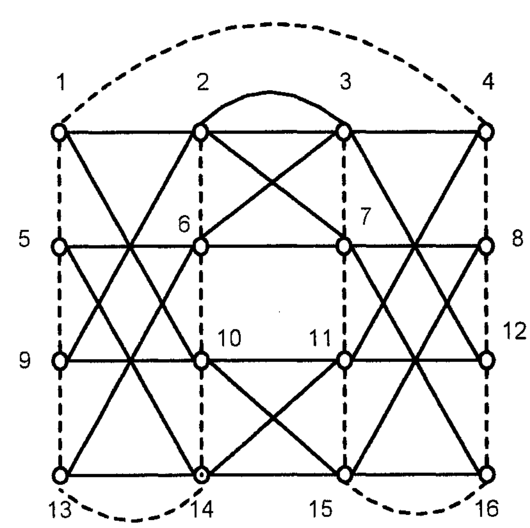


Figure 3.7: The 4-input Benes network

3.2.14 The Boolean n-Cube

This family of graphs has been proposed as desirable network architecture for a highly parallel computer. A *Boolean n-cube* $C(n)$ (or hypercube) is a graph $Q_n = (V, E)$ where n denotes its dimension. The Boolean n-cube has as vertices all binary strings of length n. The edges of $C(n)$ connect string-vertices x and y only when x and y are unit Hamming distance apart, i.e., when there exist binary strings α , β, of collectives length n - 1, such that $\{x, y\} = \{\alpha 0\beta, \alpha 1\beta\}$. Therefore, the n-cube has 2^n nodes and $n2^{n-1}$ edges. Figure 3.8 illustrates the Boolean 4-cube. **Chung et al.** [CLR87] showed that the graph $C(n)$ ($n \geq 2$) can be embedded in (n-1) pages.

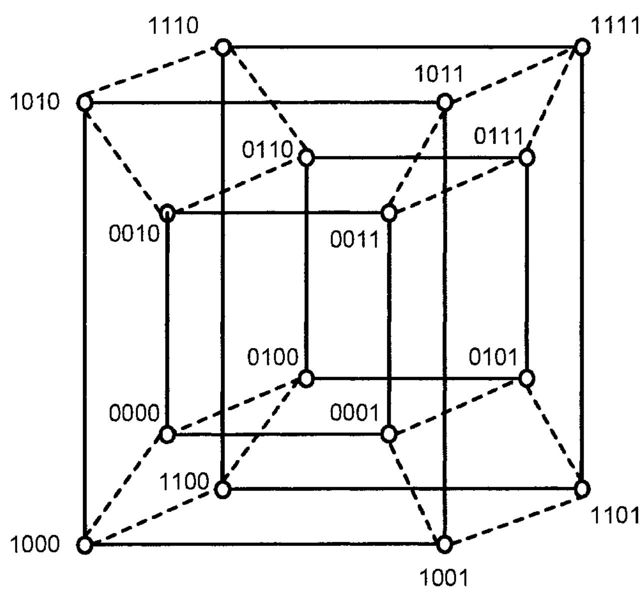


Figure 3.8: The Binary hypercubes C(4)

3.2.15 De Bruijn, Kautz and Shuffle-Exchange Graphs

The *de Bruijn digraph* denoted by $B(d, D)$ is a digraph whose vertices are the words of length D on an alphabet of d letters (written usually $\{0, 1, \ldots, d\text{-}1\}$). There is an edge from any vertex $(v_0, v_1, \ldots, v_{D-1})$ to the d vertices $(v_1, \ldots, v_{D-1}, \alpha)$, where α is any letter of

{0, 1, ..., d-1}. The *Kautz digraph*, denoted by $K(d, D)$, is a digraph whose vertices are the words of length D on an alphabet of d+1 letters with no two connective identical letters. There is an edge from any vertex (v_0, v_1, ..., v_{D-1}) to the d vertices (v_1, ..., v_{D-1}, α), where α is any letter of {0, 1, ..., d} different from v_{D-1}. The de Bruijn and Kautz digraphs represent interconnection network for massively parallel computers. The *shuffle-exchange digraph* denoted by $S(D)$ is a graph whose vertices are the 0-1 vectors of length D. There is an edge called shuffle-edge between any vertex (v_0, v_1, ..., v_{D-1}) and the vertex (v_1, ..., v_{D-1}, v_0).Also, there is an edge called an exchange-edge between any vertex (v_0, v_1, ..., v_{D-1}) and the vertex (v_0, v_1, ..., v_{D-2}, α), where $\alpha \neq v_{D-1}$. The shuffle-exchange graph is well-known as an interconnection network for parallel computers.

Obrenic [O91] showed that the binary de Bruijn and shuffle-exchange graphs can be embedded in a book of 5 pages. The best known lower bound on the pagenumber of de Bruijn and shuffle-exchange graphs is three since the graphs are nonplanar. **Hasunuma** and **Shibata** [HS97] showed that the de Bruijn digraph $B(d, D)$ and the Kautz digraph $K(d, D)$ can be embedded in (d +1) pages and the shuffle-exchange graph $S(d)$ can be embedded in 3 pages. The book embedding for $B(2, D)$, $K(2, D)$ and $S(d)$ are optimal with respect to the number of pages which follows from the nonplanarity of these networks, i.e. 3 page. However, it remains an open question whether (d + 1) pages are necessary for embedding $B(d, D)$ and $K(d, D)$ for $d > 2$.

3.2.16 Toroidal Graphs

A *toroidal graph* is the one which can be drawn on the torus with no edge crossings. A *torus* is a ring-shaped surface generated by rotating a circle around an axis that does not intersect the circle. Figure 3.9 illustrates a graph drawn on the surface of a torus [W98]. **Heath** and **Istrail** [HS92] conjectured that the pagenumber of toroidal graphs is seven although their algorithm only guarantees a 13-page book layout. Subsequently, **Endo** [E97] showed that any toroidal graph can be embedded in a book of seven pages. He also conjectured that this upper bound is tight.

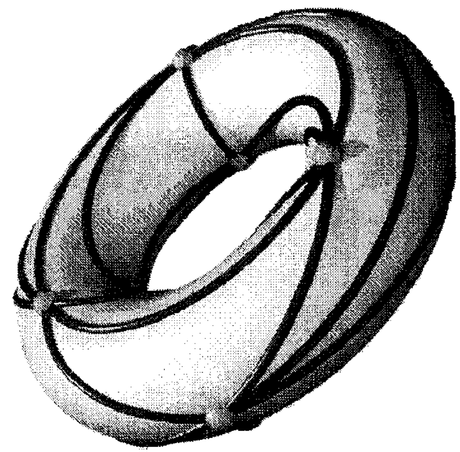


Figure 3.9: A graph drawn on the surface of a torus

3.2.17 Genus g Graphs

A surface is said to be of *genus g* if it is topologically equivalent to a sphere with *g* handles (or a doughnut with *g* holes). A *graph of genus g* is a graph which can be drawn without crossings on a surface of genus *g*, but not on one of genus $g - 1$. Conjectured by **Heath** and **Istrail** [HI92] and then proved by **Malitz** [M94b], the pagenumber of a graph of genus g is $O(\sqrt{g})$.

Since the possibility of finding a deterministic polynomial time algorithm for computing exact pagenumbers of arbitrary graphs appears to be remote, the attention was shifted to that of finding a heuristic or approximation or restricted graphs. **Kapoor et al.** [KRSZ02] presented the first genetic algorithm for solving the pagenumber problem for interconnection networks graphs such as hypercubes and meshes.

Table 3.1 summarizes some of the known upper bounds on the pagenumber of various classes of graphs discussed in this chapter. The symbol $^{+}$ indicates the exact value of the upper bound of the pagenumber.

Table 3.1: Upper bounds on the pagenumber of graphs

Graph Class	Pagenumber	Reference
outerplanar$^+$	1	[BK79]
subgraph of Hamiltonian planar$^+$	2	[BK79]
planar$^+$	4	[Y86,Y89]
series parallel (planar)$^+$	2	[CLR87]
X-tree$^+$	2	[CLR87]
square grids$^+$	2	[CLR87]
bipartite planar$^+$	2	[FMP95]
planar with quadrilateral faces$^+$	2	[FMP95]
Halin$^+$	2	[G95]
K-trees$^+$	k+1	[GH01]
toroidal	7	[E97]
Complete	$\lfloor n/2 \rfloor$	[CLR87]
genus g	$O(\sqrt{g})$	[M92b]
m edges	$O(\sqrt{m})$	[M94a]
n vertices	$\lceil \frac{n}{2} \rceil$	[BK79, CLR87]

Chapter 4

The Pagenumber of Ordered Sets

4.1 Overview

This chapter surveys the pagenumber problem for ordered sets. We discuss the known results for some restricted classes of ordered sets as well as the complexity of the pagenumber problem. We also show several interesting examples discussed widely in the literature. For basic definitions of ordered sets and its parameters, readers are advised to consult chapter 2. Some proofs are omitted and interested readers are also referred to the literature.

4.2 Literature Review

Nowakowski and **Parker** [NP89] were the first to introduce the pagenumber of an ordered set. It is the stacknumber of an ordered set's Hasse diagram viewed as a directed graph. Their first result shows that $page(P) = 1$ if and only if $cov(P)$ is a forest. Computing the pagenumber of ordered sets seems to be a very difficult question. In fact very little progress has been accomplished since the first paper on the subject [NP89]. Most of the known results relate to classes of ordered sets with a pagenumber 2, and even the question regarding a general characterization of ordered sets with pagenumber 2 is still open.

4.2.1 Ordered Sets with Pagenumber One

We call a subset C of an ordered set a *cycle* if C is a non-directed cycle in the graph *cov*(P). A cycle ordered set can be embedded in 2 pages. **Nowakowski** and **Parker** [NP89] observed that *page*(P) ≥ 2 if *cov*(P) contains a cycle.

An ordered set is P is a *forest* if *cov*(P) is a forest as a graph. The next theorem gives the pagenumber for forest ordered sets.

Theorem 4.1 (**Nowakowski** and **Parker** [NP89], **Syslo** [S89]).
Let P be an ordered set. Then, page(P) = 1 if and only if cov(P) is a forest.

Proof of Theorem 4.1.
A forests is an acyclic graph consisting only (possibly disconnected) trees. Let us assume that *cov*(P) is a tree. We can show that *page*(P) = 1 by induction on $|P|$. Since *cov*(P) is a tree, there is an element u adjacent to a unique element. Thus, either u is maximal or minimal in P. Let us assume it is maximal. Let v be the unique element adjacent to u in *cov*(P).

If $|P| = 1$ or $|P| = 2$, then *page*(P) = 1. By induction, *page*(P- $\{u\}$) = 1. *cov*($P - \{u\}$) is a tree . So, there is a one-page linear extension L' of $P - \{u\}$. We can obtain L by adding u right above v in L'. Obviously, L is a linear extension of P. We can draw the edge (v, u) in the first page without edge crossing because u $\succ$ v in L.

One can also easily show that the converse hold by showing that when *cov*(P) contains a cycle C then *page*(P) ≥ 2. A cycle can never be drawn in one page.

■

4.2.2 Complete Bipartite Ordered Sets $P_{m,n}$

The *complete bipartite* ordered set $P_{m,n}$ is an ordered set with m minimal and n maximal such that each maximal covers each minimal. The next theorem gives the pagenumber of the *complete bipartite* ordered set $P_{m,n}$.

The concept of k-twist is useful in proving lower bounds on the pagenumber for several cases. A k-edge set $\{(a_i, b_i)$: $1 \leq i \leq k\}$, in an ordered set P, forms a k-twist in a linear extension L of P, if we have in L $a_1 < a_2 < \ldots < a_{k-1} < a_k < b_1 < b_2 < \ldots. < b_{k-1} < b_k$.

We call a linear extension L of an ordered set P a *k-twist free* if there is no k-twist in L. For any L which contains a k-twist, $page(P, L) \geq k$. The existence of a twist clearly increases the pagenumber since no two edges in a twist can be assigned to the same page.

The next two theorems will use the fact that, for every ordered set P, and for every linear extension L of P used as a spine for a page embedding of P, all successors and predecessors of the same element x could be drawn on one single page. Therefore if C is a vertex cover of P (that is every covering edge has one of the end vertices in C) then $page(P) \leq |C|$.

Theorem 4.2 **(Syslo** [S89]).
The pagenumber of the complete bipartite ordered set $P_{m,n}$ *is* $min\{m, n\}$.

Proof of Theorem 4.2
Assume that $min\{m, n\} = m$. The inequality $page(P_{m,n}) \leq m$ follows from the fact that we could have a vertex cover of size m. On the other hand, it is noticed that every linear extension of $P_{m,n}$ contains an m-twist which can not be drawn on the same page. Hence, page$(P_{m,n}) \geq m$. Therefore, $page(P_{m,n}) = min\{m, n\}$

■

Figure 4.1 illustrates a complete bipartite ordered set $P_{2,3}$ and its book embedding. We can embed the ordered set in two pages since $min\{m,n\}$ is two.

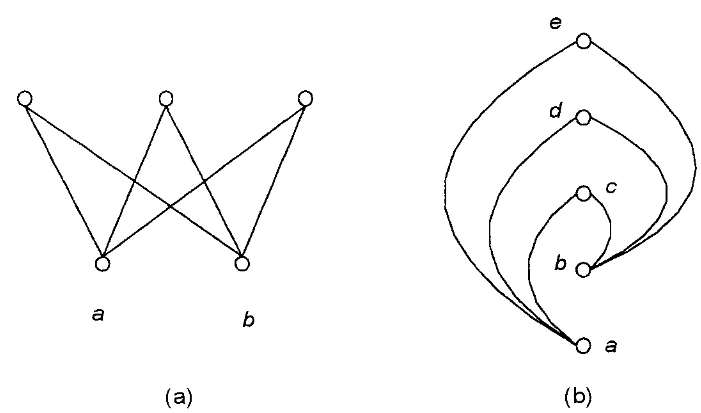


Figure 4.1: Complete bipartite ordered set

4.2.3 Complete Tripartite Ordered Sets $P_{m,k,n}$

The next result is also due to **Syslo** [S89]. Let $P_{m,k,n}$ denote a complete tripartite ordered set with m minimal elements, n maximal elements and k elements, each of which is greater than every minimal and smaller than every maximal one.

Theorem 4.3 (**Syslo** [S89]).
The pagenumber of the complete tripartite ordered set $P_{m,k,n}$ is $min(m + n, k)$.

Proof of Theorem 4.3.
We can cover all edges by either taking all minimal and maximal elements (i.e $m + n$) or by taking all the elements in the middle level (i.e. k). Therefore, $page(P_{m,k,n}) \leq min(m + n, k)$. On the other hand, all linear extensions of $P_{m,k,n}$ contain $min(m + n, k)$-twist, such that one end of the edges is on the middle level and the other end is an element on one of the other two levels.

■

4.2.4 Pagenumber of Planar Ordered Sets

Nowakowski and **Parker** [NP89] derived a general lower bound on the pagenumber of a planar ordered set and they questioned whether the pagenumber of the class of planar ordered sets is unbounded.

It is clear that for any ordered set, $page(cov(P)) \leq page(P)$, where $page(cov(P))$ is the pagenumber of the non-directed graph of the covering relations of P. For example, $page(P) = 2$ for the ordered set given in Figure 4.2 while $page(cov(P)) = 1$ for the same ordered set. Although the ordered set must be planar if $page(P) \leq 2$, the converse is not always true. Several planar ordered sets with pagenumber 3 are illustrated in Figure 4.3. These ordered sets are due to **Nowakowski**, **Kelly** and **Czyzowicz**. Notice that, Figure 4.3(c) illustrates a three-page planar ordered set of smallest size.

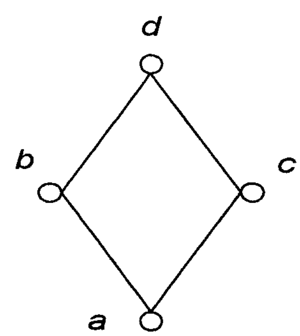


Figure 4.2: Diamond ordered set

Dilworth [K77] enumerated all 7-elements planar ordered sets that were noted to have a pagenumber of 2 [A96]. Consequently, every planar ordered set with seven elements or less has pagenumber two.

Hung [H93] showed that there exists a 48-element planar ordered set with pagenumber 4. This ordered set, shown in Figure 4.4, is the smallest known 4-page planar ordered set. It was used by **Hung** [H93] to create a family of simple planar ordered sets which requires 4 pages. Figure 4.5 illustrates a 4-page book-embedding of the planar ordered set given in Figure 4.4. An edge drawn as a solid line to the left (right) of the spine is assigned to page 1 (page 3), and an edge drawn as a dashed line to the left (right) of the spine is assigned to page 2 (page 4).

However, no one is aware of any planar ordered set with pagenumber of 5. It remains an unsettled question how many pages are needed for planar ordered sets.

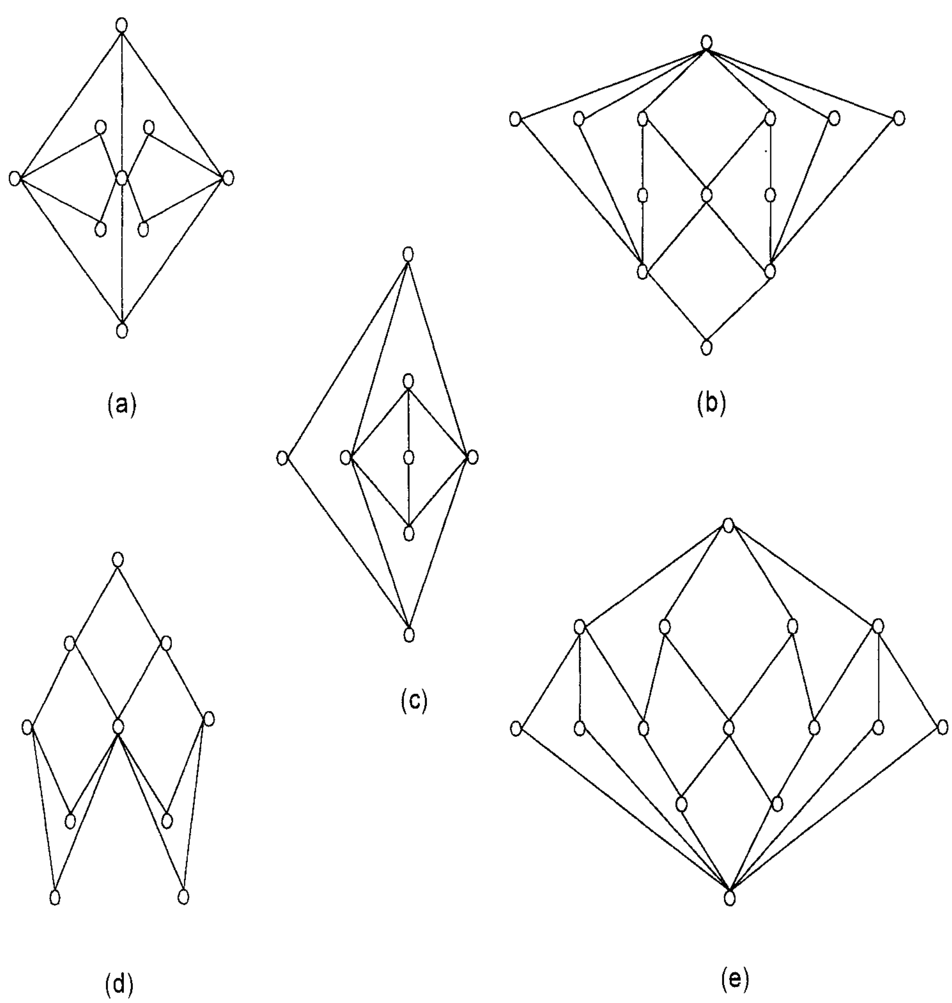


Figure 4.3: Planar ordered sets of pagenumber 3

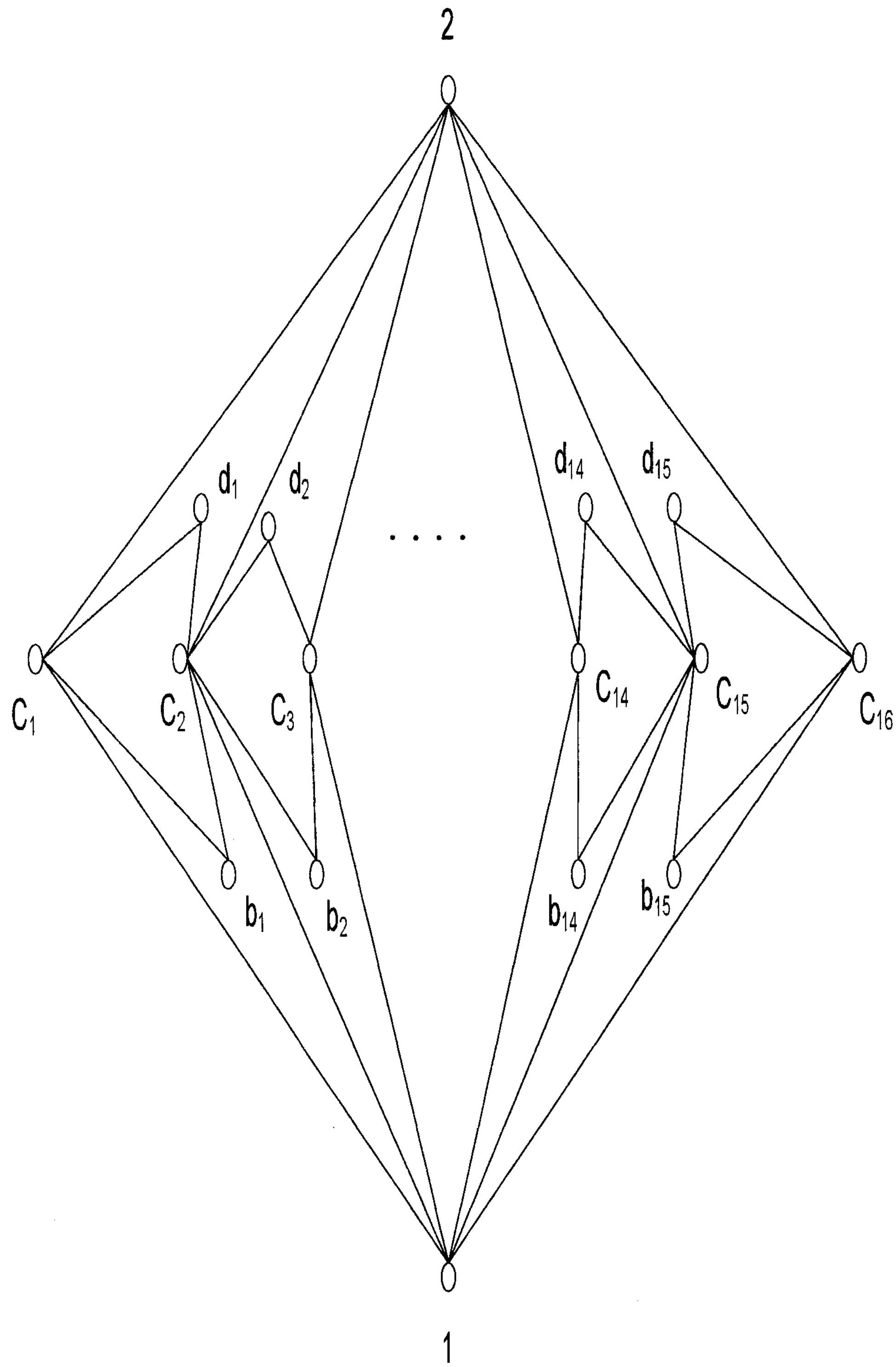


Figure 4.4: A 48-Element planar ordered set of pagenumber 4

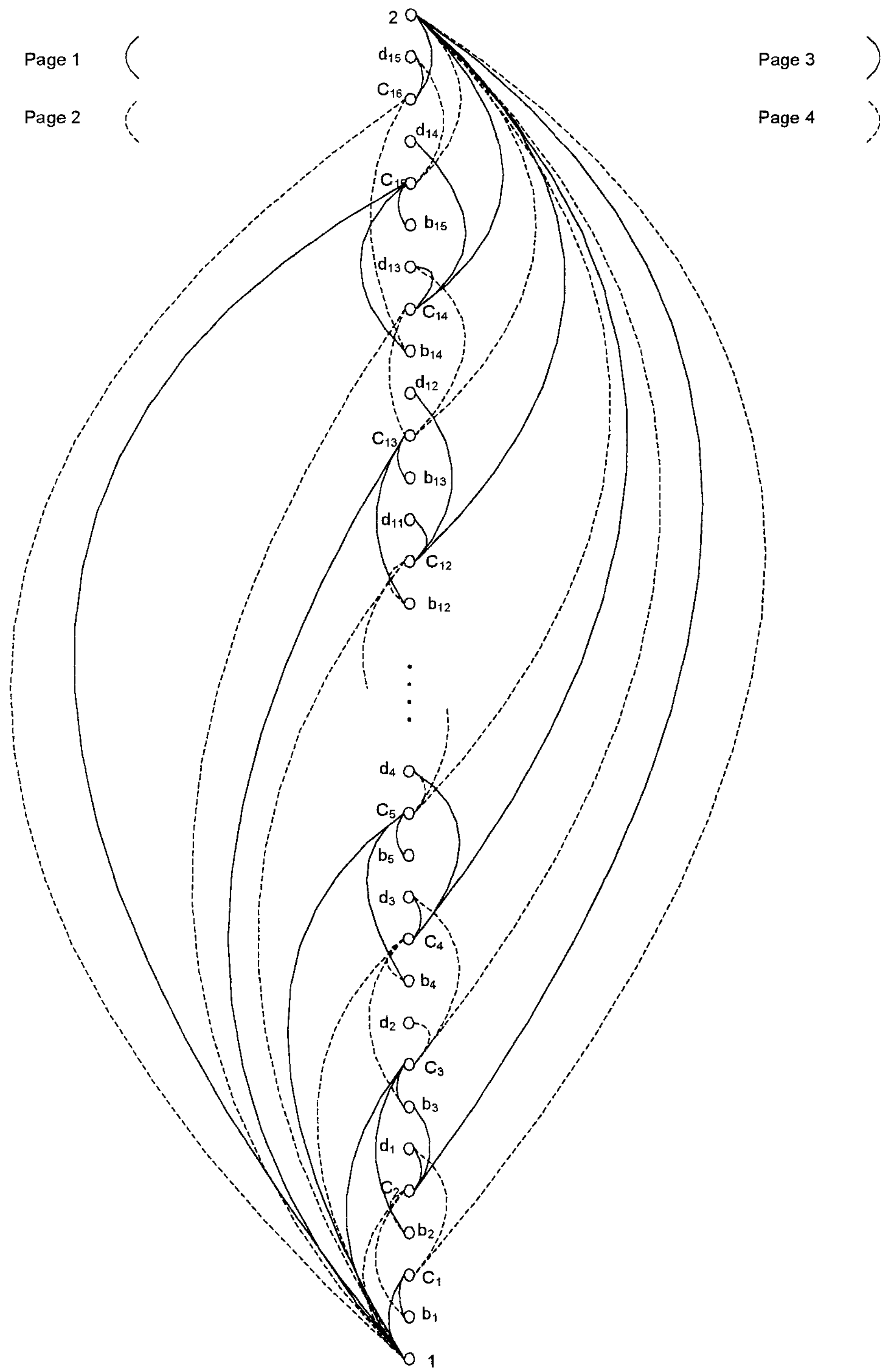


Figure 4.5: 4-Page embedding of a planar ordered set

4.2.5 Series Parallel Planar Ordered sets

The first important class of planar ordered sets for which we have a complete answer is the class of series parallel planar ordered sets. Recall that a *series-parallel ordered set* is an ordered set P that can be constructed from singletons using only the constructions of disjoint sum and linear sum.

Theorem 4.4 (**Alzohairi** and **Rival** [AR96]).
If P is a series parallel planar ordered set, then page(P) ≤ 2.

For an ordered set P, the *completion* $\overline{P}$ of P is the smallest lattice into which P can be embedded. Notice that if P is a planar series parallel ordered set then $\overline{P}$ is a planar series parallel lattice.

The general idea of the proof for this theorem works as follows. First, it was shown that every series parallel planar lattice can be embedded into two pages [A96]. Using the 2-page embedding of the completion, a 2-page embedding of the original ordered set was deduced. This was done by removing those elements introduced during the completion. This construction led to an $O(n^3)$ algorithm for the two page embedding of series parallel planar lattices, where n is the number of the elements of the lattice. Figure 4.6 illustrates a series parallel planar lattice with its two-page book embedding.

Notice, that the linear extension L generated by the polynomial algorithm in [AR96] is the left greedy linear extension (i.e., if $x \parallel y$ in P, and y lies to the left of x, then $y < x$ in L).

Giacomo et al. [GDLW02] presented the first linear time $O(n)$ algorithm to draw an upward two-page book embedding of a planar series parallel digraph even with transitive edges.

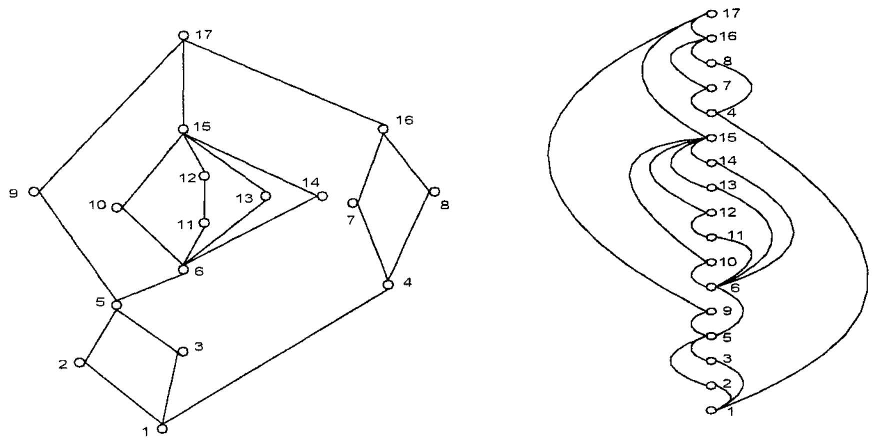


Figure 4.6: A series parallel planar lattice with its book embedding

4.2.6 Spherical Ordered Sets.

An ordered set is spherical if it has an upward drawing on the surface of the sphere such that all arcs are strictly increasing northward on the sphere, and no pair of arcs cross. Figure 4.6 illustrates the 2^3 spherical ordered set [A96]. **Alzohairi et al.** [ARK01] showed that the pagenumber of spherical lattices is unbounded.

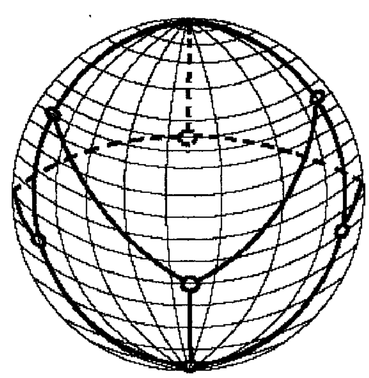


Figure 4.7: The 2^3 spherical ordered set

4.2.7 Lower Bounds

Nowakowski and **Parker** [NP89] provided a lower bound for the pagenumber of a planar ordered set P in terms of the number of its edges $e(P)$, vertices $v(P)$ and its height.

Theorem 4.5. (**Nowakowski** and **Parker** [NP89]).
Let P be a planar ordered set, if page(P) > 1 then

$$page(P) \geq \begin{cases} \dfrac{e(P)}{v(P)-1-height(P)/2} & \text{if height(P) is even} \\ \dfrac{e(P)-2v(P)+2+height(P)}{v(P)-1-(height(P)-1)/2} & \text{if height (P) is odd} \end{cases}$$

The proof of Theorem 4.5 is a consequence of the following result [NP89] *for planar ordered sets,*

$$e(P) \leq 2|P| - 2 - height(P).$$

A finite ordered set is graded of rank n if every maximal chain has length n. The *pagenumber* of a planar lattice is not yet known. However, the pagenumber of a planar graded lattice is at most two (**Nowakowski** and **Paker** [NP89]).

Syslo [S89] provided a lower bound on the pagenumber of an ordered set in terms of its jumpnumber and bumpnumber. He showed that ordered sets with jumpnumber 1 have a pagenumber at most 2. He also showed that ordered set with jumpnumber 2 can have an arbitrarily large pagenumber.

Every linear extension L of an ordered set P can be expressed as the linear sum of chains of P, i.e.

$$L = C_0 \oplus C_1 \oplus \ \ldots C_k$$

where C_i $(0 \leq i \leq k)$ is a chain of P and the top of C_i is noncomparable in P with the bottom of C_{i+1} for $0 \leq i \leq k-1$. The number of non-comparabilities between consecutives elements in L (i.e k) is called the jumpnumber of P with respect to L.

Theorem 4.6 (**Syslo** [S89])
Let P be an ordered set with n elements and m edges and let $L = C_0 \oplus C_1 \oplus \ \ldots C_k$ be a linear extension of P. Then

$$page(P, L) \geq \left\lceil \frac{m-(n-1-k)}{k} \right\rceil$$

Proof Theorem 4.6. In the chain decomposition (see previous page) of an ordered set P, each page does not contain more than one edge between elements of any two chains C_i and C_j where $i < j$. We cannot also have a cycle on one page produced by some spine edges and nonspine edges drawn on that page. So, if L consists of $k + 1$ chains, then a page may contain at most k nonspine edges of P. Therefore, the number of spine edges is equal to $n - 1 - k$.

■

As consequence, **Syslo** [S89] also derived the next corollary which relates the pagenumber to the bumpnumber of an ordered set P. The *bumpnumber, bump*(P) of P is equal to maximum number of *jump*(P, L).

Corollary 4.1 (**Syslo** [S89]).

Let P be an ordered set, then page(P) $\geq \left\lceil \frac{m-n+1}{bump(P)} \right\rceil + 1.$

Table 4.1 summarizes some of the known bounds on the pagenumber of various restricted classes of ordered sets discussed in this chapter. The symbol $^+$ indicates the exact value of the upper bound of the pagenumber.

Table 4.1: Upper bounds on the pagnumber of ordered sets

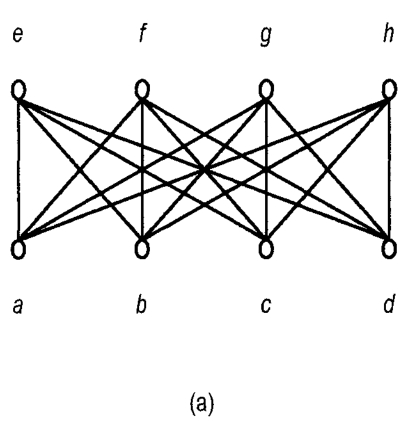
Ordered Set Class	Pagenumber	Reference
forest$^+$	1	[NP89, S89]
jumpnumber 1$^+$	2	[S89]
complete bipartite $K_{m,n}$ $^+$	*min* $\{m, n\}$	[S89]
series parallel planar$^+$	2	[AR96]
series parallel planar lattice$^+$	2	[AR96]
planar graded lattice$^+$	2	[NP89]
lattice of width two$^+$	2	[AR96]

4.2.8 The Pagenumber and Order Dimension

The *order dimension* of an ordered set P is the smallest number k of linear extensions L_1 $L_2, \ldots, L_k$ whose intersection is P, that is, $x \leq y$ in P if and only if, for every $i = 1, 2, \ldots, k$, $x \leq y$ in L_i. The order dimension of the ordered set $P_{4,4}$ shown in Figure 4.8(a) is two. ($P = L_1 \cap L_2$ where $L_1 = \{a < b < c < d < e < f < g < h\}$ and $L_2 = \{d < c < b < a < h < g < f < h\}$).

Alzohairi [A96] showed that the pagenumber for an ordered set is not related to its order dimension. In fact, the pagenumber of an ordered set of dimension two is not bounded. We can find a sequence of two dimensional ordered sets with unbounded pagenumber. For example, the pagenumber of the complete bipartite ordered set $P_{n,n}$ with n minimal and n maximal shown in Figure 4.8(b) is n ($page(P_{n,n}) = n$) for each positive integer n while its order dimension is two. ($P = L_1 \cap L_2$ where $L_1 = \{ 1 < 2 < \ldots < n-1 < n < n+1 < \ldots < 2n-1 < 2n\}$ and $L_2 = \{ n < n-1 < \ldots < 2 < 1 < 2n < 2n-1 < \ldots < n+2 < n+1\}$).

However, the pagenumber for an ordered set may relate to its order dimension if it is restricted to planar ordered sets of dimension two such as planar lattices [A96].



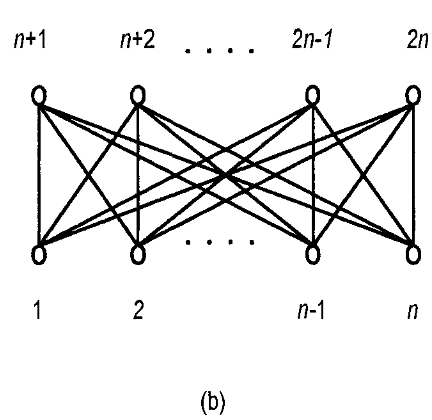


Figure 4.8: Ordered set $P_{4,4}$ (a) and Ordered set $P_{n,n}$ (b)

4.2.9 Complexity of the Pagenumber Problem

Lenwood et al. [HP99] showed that the problem of determining whether a directed acyclic graph (DAG) can be embedded in 6-pages is NP-complete. **Alzohairi** [A96] showed that the problem of determining the minimum number of pages required for a fixed linear extension of an ordered set is also NP-complete.

In this section we provide a formal proof of the complexity of finding the pagenumber for a fixed linear extension of an ordered set.

Equivalent formulation is often more helpful for analysis. We can view the ordering of the spine as ordering of the vertices on a circle, and then the pages are collection of noncrossing chords. Let P be an ordered set and L be a linear extension of P. Given a graph $G = (V, E)$ and a permutation δ of the vertices of G, a layout of G with respect to δ is a drawing of G on a page in which vertices are listed on a line segment at unit intervals according to δ and edges as concave arcs in such a way that two edges intersect if necessary. Figure 4.9 shows an ordered set with its layout graph.

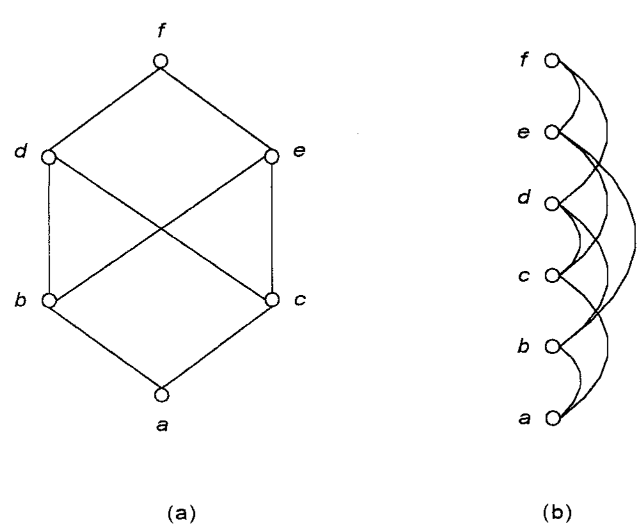


Figure 4.9: A layout graph

Let us define the *intersection graph* of the layout graph of *G* with respect to δ as a new graph whose vertices are the edges of *G*, and two edges correspond to edges in the intersection graph if they intersect in the layout graph of G with respect to δ. Figure 4.10 shows the intersection graph of the layout graph *cov*(*P*) shown in Figure 4.9 with respect to the given linear extension.

The *chromatic number* of a graph G is the minimum number of colors needed to color the vertices of G such that no adjacent vertices are assigned the same color. Theorem 4.7 connects the pagenumber problem to the chromatic number problem.

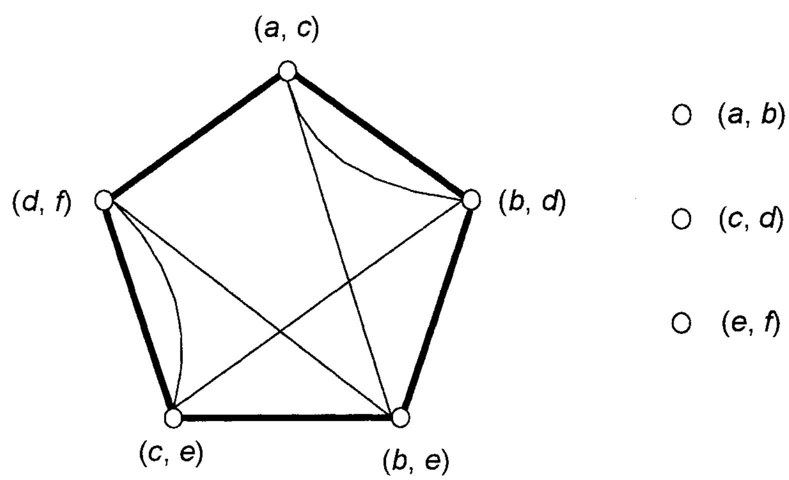


Figure 4.10: Intersection graph

Theorem 4.7 (Even and Itai [EI71]).

Let P be an ordered set and L a linear extension of P. Then page(P, L) is equal to the chromatic number of the intersection graph of the layout graph of cov(P) with respect to L.

Proof of Theorem 4.7.

Suppose that the chromatic number of the intersection graph of the layout of *cov*(*P*) with respect to *L* is equal to *m*.

i. $page(P, L) \geq m$. Suppose $page(P, L) = n$ and let E_i be the set of edges in the i-th page, $1 \leq i \leq n$. Since edges in the same page do not cross, there are no two adjacent vertices of the intersection graph of the layout graph of $cov(P)$ with respect to L, in the set E_i. So, we can color E_i by the color i, for each i. Hence, $m \leq n$ and $n = page(P, L)$

ii. $page(P, L) \leq m$. Let E_i be the set of edges colored by the i-th color $1 \leq i \leq t$. We can draw the set of edges E_i in the i-th page for each $i = 1, \ldots, m$ because in the layout graph of $cov(P)$ with respect to L, no two edges of the set E_i cross if they are drawn in the same page. Hence, $page(P, L) \leq m$.

■

Corollary 4.2 ([A96]).

Let $k \geq 3$ be an integer. The decision question whether, for a given ordered set P and an arbitrary linear extension L of P, page(P,L) $\leq$ k, is NP-complete. In fact, this decision question is NP-complete even if P is bipartite.

Proof of Corollary 4.2.

Let P be an ordered set and L be a linear extension of P. Let H be the intersection graph of the layout of $cov(P)$ with respect to L. In H consider each edge (x, y) as an interval in the real line. Thus, H is an overlap graph, where a graph K is called an overlap graph if its vertices may be put into one-to-one correspondence with a collection of intervals on a line such that two vertices are adjacent in K if and only if their corresponding intervals overlap and not just intersect.

It is known that an undirected graph is an *overlap graph* if and only if it is a circle graph. A *circle graph* is a graph whose vertices are the chords of a cycle in which two vertices form an edge if the two chords cross. Figure 4.11 illustrates a circle graph and its representation by a set of chords.

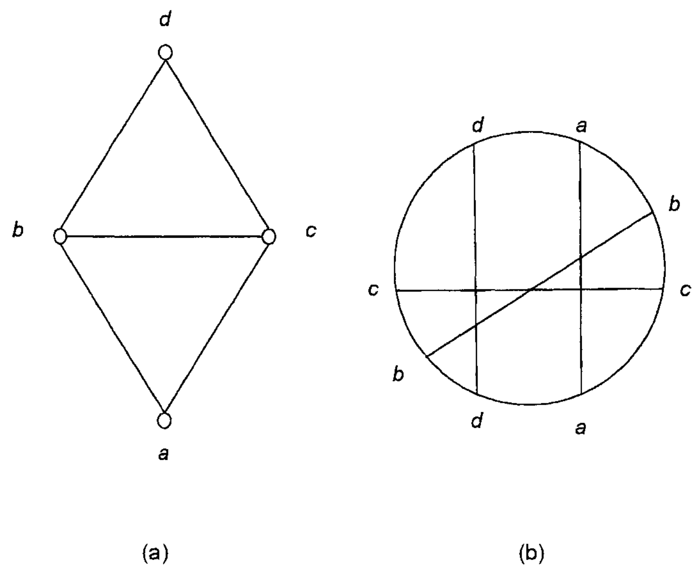


Figure 4.11: A circle graph

The problem of minimizing the number of pages that L requires is equivalent to coloring circle graphs according to Theorem 4.7. Thus, it is NP-complete [GJMP80].

Suppose H is a circle graph. Then, H is an overlap graph. We may assume that the intervals are either open or closed and that no two intervals have a common end point.

We can accomplish all of these constructions and transformations in polynomial time. We now want to construct a bipartite ordered set P and a linear extension L of P, such that the intersection graph of the layout of $cov(P)$ with respect to L is H. Let the elements of P be endpoints of the intervals which are ordered, for each interval I, by $sup(I) \succ inf(I)$. Then, order the elements in L as they are ordered in the real line.

The proof is still valid even if it is restricted to the class of bipartite ordered sets [A96].

■

Chapter 5

New Results on the Pagenumber

5.1 Overview

This chapter discusses the main contributions of the thesis. As previously mentioned (see section 4.2.9), the pagenumber of ordered sets was proven to be NP-complete even for a fixed linear extension. Therefore, we cannot expect to find a polynomial algorithm. Instead, one can either provide a heuristics approach or an algorithm that works for special classes of ordered sets.

In this chapter we present our main contributions including:

1. The pagenumber of bipartite interval orders.
2. An upper bound for the pagenumber of bipartite ordered sets.
3. An upper bound for the pagenumber of complete multipartite ordered sets with length 4 and 5.

5.2 The Pagenumber of Interval Orders

This section discusses the pagenumber of bipartite and tripartite interval orders.

5.2.1 The Pagenumber of Bipartite Interval Orders

In this section, we prove that the pagenumber of a bipartite interval order P is equal to the maximum pagenumber of a complete suborder of P. We use a technique that relies on easily identifying complete suborders within a given bipartite interval order. An algorithm for finding the pagenumber of bipartite interval orders is deduced.

A *bipartite interval order P* has two disjoint sets of elements, *M* and *N*, with the property that each covering relation in *P* has one element in *M* and one element in *N*. The *complete bipartite interval order* $P_{m,n}$ is obviously a special case of interval orders with *m* minimal and *n* maximal. Figure 5.1 depicts a bipartite interval order P. The highlighted edges with bold represent a complete suborder P' of the interval order.

We know that for any interval order *P* the following two conditions are satisfied:

- The sets of predecessors are linearly ordered with respect to inclusion.
- The sets of successors are also linearly ordered with respect to inclusion.

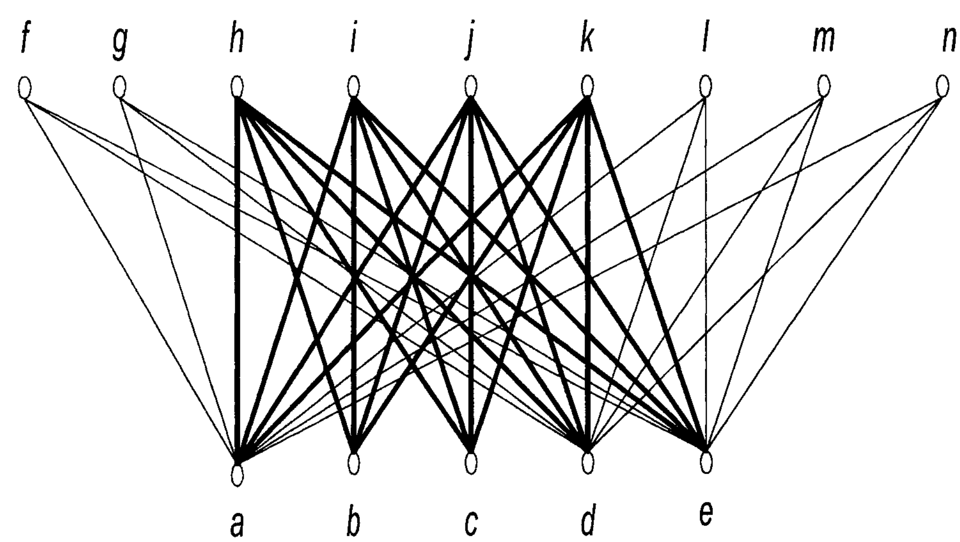


Figure 5.1: A bipartite interval order

Theorem 5.1. *The pagenumber of a bipartite interval order is equal to the maximum pagenumber of complete suborders of P.*

In order to prove Theorem 5.1 we need a sequence of lemmas. Let $P = (M, N)$ be a bipartite interval order with a set M of minimal elements of size m and a set of N of maximal elements of size n. Let $M = (m_1, m_2, \ldots, m_m)$ be the list of minimal elements of P arranged in a decreasing order with respect to the inclusion relation of the sets of successors, i.e. $Succ(m_1) \supseteq Succ(m_2) \supseteq \ldots Succ(m_m)$. Let $N = (n_1, n_2, \ldots, n_n)$ be the list of maximal elements of P arranged in decreasing order with respect to the inclusion relation of the sets of predecessors, i.e. $Pred(n_1) \supseteq Pred(n_2) \supseteq \ldots Pred(n_n)$.

Note that $page(P') \leq page(P)$ if P' is a complete suborder of P.

Lemma 5.1. *Let $P = (M, N)$ be a bipartite interval order, and let $P' = (M', N')$ be a complete suborder of P. Then there exists i and j such that $M' \subseteq \{m_1, m_2, \ldots, m_i\}$ and $N' \subseteq \{n_1, n_2, \ldots, n_j\}$ and $(\{m_1, m_2, \ldots, m_i\}, \{n_1, n_2, \ldots, n_j\})$ is a complete bipartite suborder of P.*

Proof of Lemma 5.1.

Let j be the maximum key such that $n_j \in N \cap N'$(since $N' \subseteq N$). Since $Pred(n_j) \subseteq Pred(n_k)$ for every $k \leq j$ then $(M', \{n_1, n_2, \ldots, n_j\})$ is a complete suborder of P. Likewise, Let i be the maximum key such that $m_i \in M \cap M'$(since $M' \subseteq M$). Since $Succ(m_i) \subseteq Succ(m_k)$ for every $k \leq i$ $(\{m_1, m_2, \ldots, m_i\}, \{n_1, n_2, \ldots, n_j\})$ is a complete bipartite suborder of P. Moreover it is obvious that $M' \subseteq \{m_1, m_2, \ldots, m_i\}$ and $N' \subseteq \{n_1, n_2, \ldots, n_j\}$

■

Lemma 5.1 shows that in order to find a complete bipartite suborder of P with the maximum pagenumber, it is sufficient to look at those ones with the structure $(\{m_1, m_2, \ldots, m_i\}, \{n_1, n_2, \ldots, n_j\})$, for some i and j. Henceforth, $P(i, j)$ will be used to refer to these complete bipartite suborders.

Lemma 5.2*. Let $P = (M, N)$ be a bipartite interval ordered set, and let $P_l = P(i, j)$ be a complete suborder of P with a maximal pagenumber.*

If $j > i$ then for every k, $l > i$, n_k is non comparable to m_l in P.

If $j < i$ then for every k, $l > j$, n_k is non comparable to m_l in P.

Proof of Lemma 5.2

Suppose that $j > i$ therefore $page(P_1) = min\{i, j\} = i$. Suppose that $n_k > m_l$ in P for some key k and $l > i$. Since $Succ(m_l) \subseteq Succ(m_r)$ for every $r \leq l$, $Succ(m_{i+1})$ will contain $\{n_1, n_2, \ldots, n_j\}$ and therefore $P_2 = P(i+1, j) = (\{m_1, m_2, \ldots, m_i, m_{i+1}\}, \{n_1, n_2, \ldots, n_j\})$ is a complete suborder of P and where $page(P_2) = min\ \{i+1, j\} = i+1 > page(P_1)$ [Since $i < j$.]. This contradicts the choice of P_1. The same argument will apply if $j < i$.

■

Proof of Theorem 5.1.

Consider a complete suborder $P' = P(i, j)$ of P with a maximum pagenumber. We prove that the bipartite interval ordered set P can also be embedded in $page(P')$ which is equal to the minimum of the two levels i.e. i, j. The optimal layout of P is obtained using the following linear extension L of P:

$$m_m < m_{m-1} < \ldots < m_1 < n_n < n_{n-1} < \ldots < n_1.$$

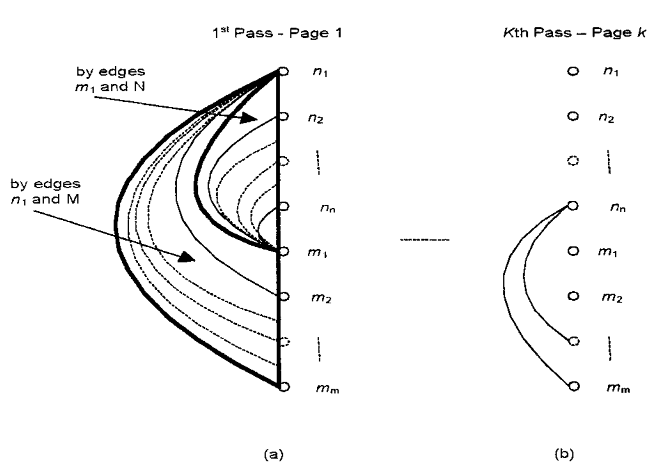


Figure 5.2: An illustration for Theorem 5.1

Without loss of generality, we may assume that $i < j$. We will use the first page to draw all covering relations of m_1 and of n_1. Clearly, we can fit all these covering relations within the first page. This is possible because the covering relation between m_1 and n_1 will leave an empty hole that we could utilize to embed the relations between n_1 and the elements in M (see Figure 5.2(a)). We continue to draw the covering relations between m_2 and n_2 in the second page and so on.

Overall, we will be able to draw all of the left cover relations of the elements m_k and n_k on the k-th page for $k \leq i$. Lemma 5.2 guarantees that all the covering relations of P will be drawn in one of the pages after i iterations (see Figure 5.2(b)).

■

Description of the Algorithm.

We will divide the algorithm to two stages, namely preprocess stage and drawing stage:

1. Preprocess Stage. The Preprocess stage composes from the following steps:
    a. Assign each element in P to either the minimal set $Min\{\}$ or the maximal set $Max\{\}$.
    b. Compute the cardinality of each element of P in $Min\{\}$ and $Max\{\}$.
    c. Sort the list of minimal elements of P in a decreasing order with respect to the inclusion relation of the sets of successors (i.e. $Min\{\} = \{m_1, m_2, \ldots, m_m\}$, and sort the list of maximal elements of P in decreasing order with respect to the inclusion relation of the sets of predecessors (i.e. $Max\{\} = \{n_1, n_2, \ldots, n_n\}$).

2. Drawing Stage. It composes from the following steps:
    a. Draw all elements of P on the spine according to the optimal linear extension of P (i.e L: $m_m < m_{m-1} < \ldots < m_1 < n_n < n_{n-1} < \ldots < n_1$).

b. Draw all covering relations incident to first minimal (m_1) in the first page. Similarly, draw all covering relations incident to the first maximal (n_1) in the first page. The page is then incremented as long as there are more covering relations to be drawn. At each step, we deal with m_i and n_i in a similar way. Supported by Theorem 5.1, the algorithm stops when either m_i or n_i has no further relation to draw. In that case, both m_i and n_i should not have further relation to draw.

Algorithm 5.1 shows a pseudo code of the drawing stage of the bipartite interval orders pagenumber drawing algorithm.

Analysis of the Algorithm Complexity.

We start by computing the complexity of the preprocess stage and the drawing stage:

1. Preprocess Stage. The Preprocess stage composes from the following steps:

   a. O(n) operations are required to assign each element of P to either the minimal set *Min*{} or the maximal set *Max*{}. O(n^2) operations are needed to assign all the elements of P to either *Min*{} or *Max*{}.
   b. To check the cardinality of each element of P in *Min*{} or *Max*{}, additional O(n) operations are required. O(n^2) operations are needed to check all elements of P.
   c. O(n log n) operations are needed to sort the list of minimal elements of P in a decreasing order with respect to the inclusion relation of the sets of successors (i.e. *Min*{} = {m_1, m_2, …, m_m}, and to sort the list of maximal elements of P in decreasing order with respect to the inclusion relation of the sets of predecessors (i.e. *Max*{} = {n_1, n_2, …, n_n}).

   In total, O(n^2) operations are needed to preprocess the bipartite ordered set P.

2. Drawing Stage. It composes from the following steps:

    a. The assignment of all elements of P according to the optimal linear extension of P (i.e L: $m_m < m_{m-1} < \ldots < m_1 < n_n < n_{n-1} < \ldots < n_1$) to the spine requires O(n) operations.

    b. Notice that, at each step, we deal with m_i and n_i. Supported by Theorem 5.1, the algorithm stops when either m_i or n_i has no further relation to draw. In that case, both m_i and n_i should not have further relation to draw. Each while statement needs O(n) operations (see Algorithm 5.1). Thus, the edge drawing step requires O(n^2) time.

    In total, O(n^2) operations are needed for the drawing stage of the algorithm.

Therefore, the algorithm has a polynomial complexity of O(n^2).

In Figure 5.1, the complete suborder is $P'_{5,4}$ and consequently can be embedded in 4 pages. Likewise, the bipartite interval order can be embedded in 4 pages with the following linear extension

$$L: b < c < e < a < d < n < m < l < g < f < k < j < i < h.$$

Figure 5.3 shows the 4-page embedding for the bipartite interval order in Figure 5.1 generated by our algorithm. We note that the first page always has the highest pagewidth and it decreases on the subsequent pages.

Algorithm (P: Bipartite Ordered Set; Min{}; Max{})

/* Input = A bipartite interval order P */

/* Output = A pagenumber drawing of P */

/* $Min\{\} = \{m_1, m_2, \ldots, m_m\}$ */

/* $Max\{\} = \{n_1, n_2, \ldots, n_n\}$ */

/* i = index to reference into $Min\{\}$ */

/* j = index to reference into $Max\{\}$ */

Begin

$i = 1$

$j = 1$

pageno = 0

Draw L on the spine such that $m_m < m_{m-1} < \ldots < m_1 < n_n < n_{n-1} < \ldots < n_1$

while ($m_i \prec n_j$ is not drawn AND $i \leq m$ AND $j \leq n$)

increment *pagno*

while ($e \in \{ n_j , n_{j+1, \ldots,} n_n\}$ such that $m_i \prec$ e AND $m_i \prec e$ is not drawn)

Draw edge $m_i \prec e$ on *pageno*

Mark $m_i \prec e$ as drawn

while ($e \in \{ m_{j+1, \ldots,} m_m \}$ such that e $\prec n_j$ AND e $\prec n_j$ is not drawn)

Draw edge e $\prec n_j$ on *pageno*

Mark e $\prec n_j$ as drawn

increment i and j

End

Algorithm 5.1: The bipartite interval orders pagenumber drawing algorithm

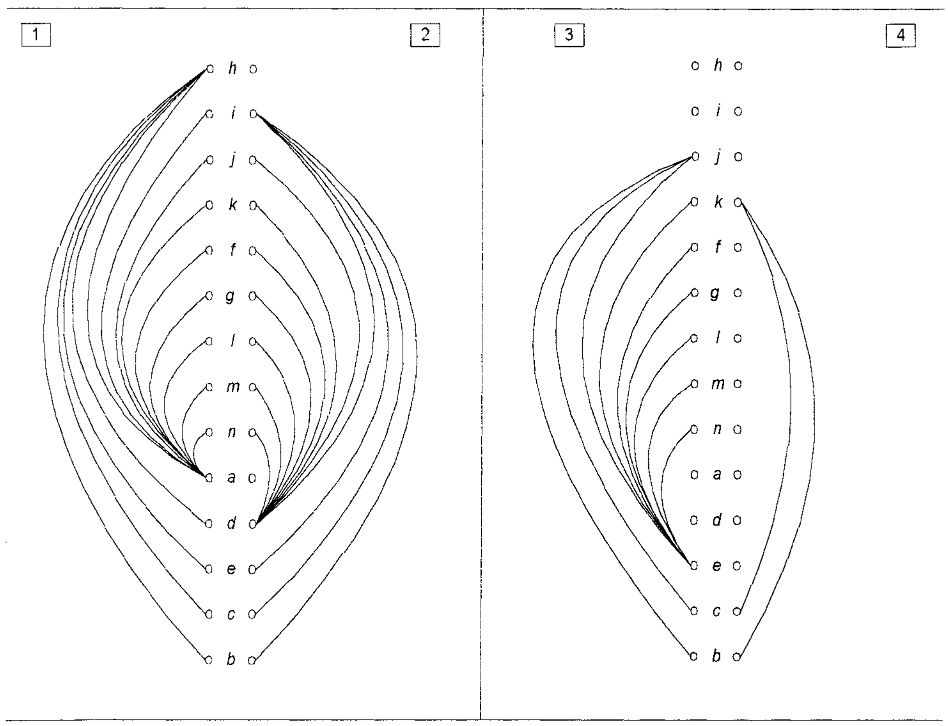


Figure 5.3: 4-page embedding of the bipartite interval order depicted in Figure 5.1

5.2.2 The Pagenumber of Tripartite Interval Orders

The case of n-partite interval order seems to be more complex than the bipartite case. The result in Theorem 5.1 cannot be extended and we do not see a generalization for the n-partite interval order and not even for the tripartite case. In fact, there are tripartite interval orders with pagenumber larger than the pagenumber of any complete tripartite suborder. For instance, Figure 5.4 illustrates a tripartite interval order P which has *page*(P) = 4 although there are no complete tripartite suborders with pagenumber larger than 3. P' is a complete tripartite suborder obtained from P with a maximum pagenumber. The page (book) layout of P and P' are illustrated in Figure 5.5 and Figure 5.6 respectively.

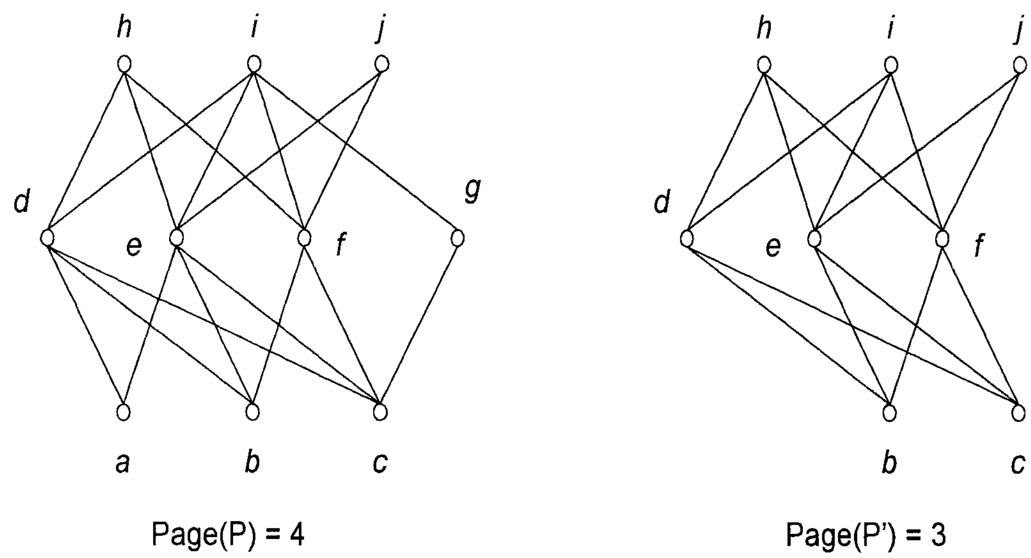


Figure 5.4: Interval order *P* with a complete multipartite suborder of *P*

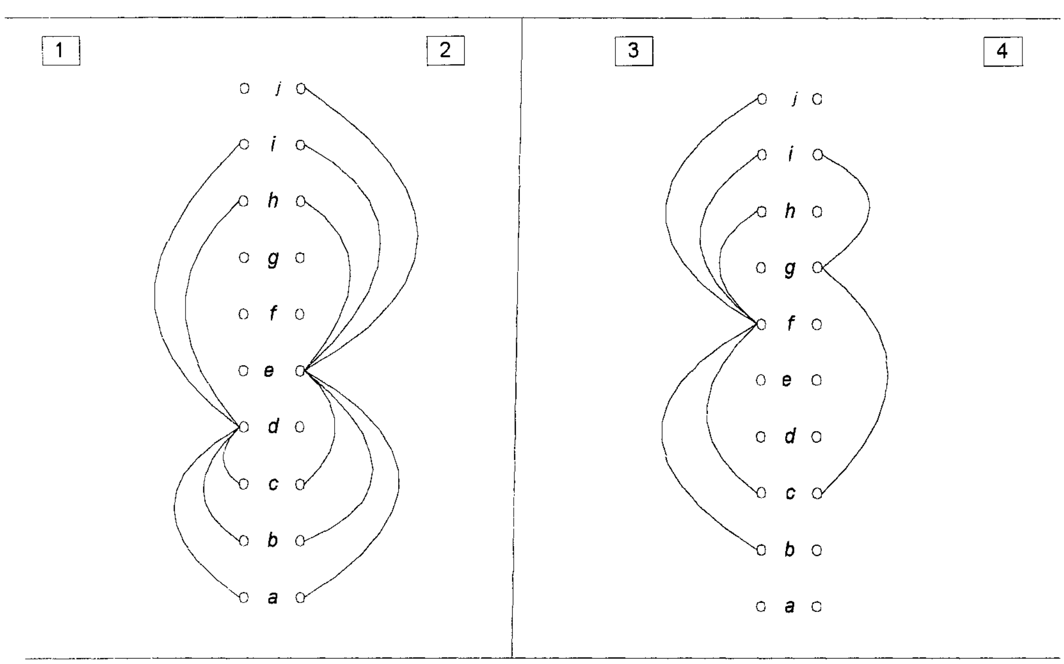


Figure 5.5: The pagenumber layout of *P* shown in Figure 5.4

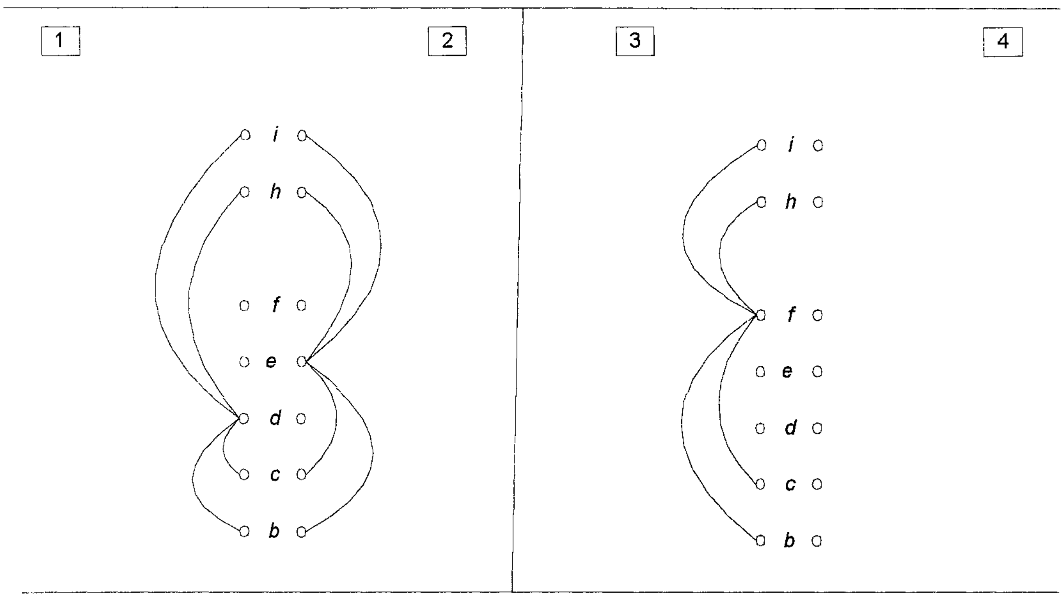


Figure 5.6: The pagenumber layout of *P*' shown in Figure 5.4

5.3 An Upper Bound for the Pagenumber of Bipartite Ordered Sets

The strategy we used in section 5.2 turns to be helpful in finding an upper bound for the pagenumber of a larger class of bipartite ordered sets. Therefore, we can utilize a similar strategy to assist us in finding an upper bound for the pagenumber of bipartite ordered sets.

We define a *zig-zag* Z in P as a partition into disjoint subsets of $M = M_1, M_2, \ldots, M_n$ and $N = N_1, N_2, \ldots, N_n$ such that

$Succ(M_i) \subseteq N_i \cup N_{i+1}$ for $1 \leq i < n$ and $Succ(M_n) \subseteq N_n$ and

$\text{Pred}(N_i) \subseteq M_{i-1} \cup M_i$ for every $1 < i \leq n$ and $\text{Pred}(N_1) \subseteq M_1$

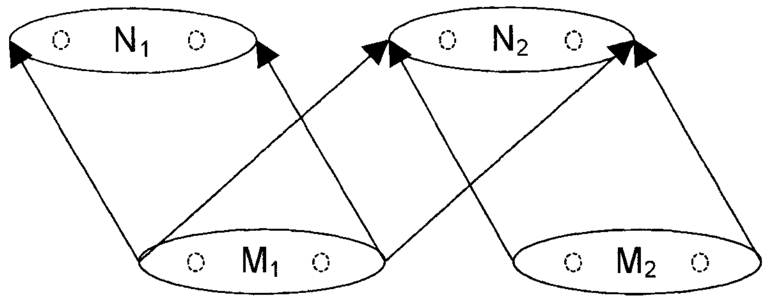


Figure 5.7: A zig-zag of length 4

The zig-zag Z has a length of $2n$ (see Figure 5.7).

Theorem 5.2. *let $P = (M, N)$ be a bipartite ordered set. Let $Z = M_1, M_2, N_1, N_2$, be a zig-zag of length 4 that covers P. Then, the pagenumber of P is bounded by the maximum value of $|M_1|$ and $|N_2|$.*

Proof. Let us enumerate all the elements of the disjoint sets of the zig-zag namely, M_1, M_2, N_1 and N_2 in random orders:

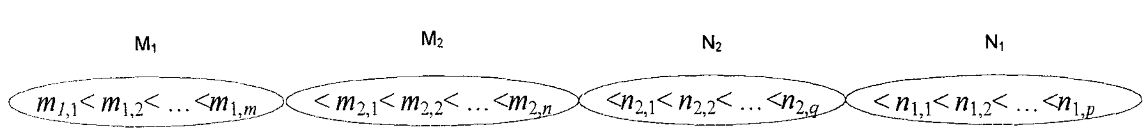
M_1: $m_{1,1} < m_{1,2} < \ldots < m_{1,m}$

M_2: $m_{2,1} < m_{2,2} < \ldots < m_{2,n}$

N_1: $n_{1,1} < n_{1,2} < \ldots < n_{1,p}$

N_2: $n_{2,1} < n_{2,2} < \ldots < n_{2,q}$.

Let us assume L is a linear extension of P obtained by enumerating the elements of P using the following ranking order:



We can utilize the first page to draw all covering relations of $m_{1,1}$ and $n_{2,1}$ as follows. It is possible to to accommodate all these covering relations within the same page (1st page) since area between $m_{1,1}$ and $n_{2,1}$ is not used (we may have a covering relation between these two vertices) and thus could be used to draw the relations between $n_{2,1}$ and the elements in M.

After k iteration such that $k \leq \max \{|M_1|, |N_2|\}$, we will be able to embed all of the remaining covering relations of the elements $m_{1,k}$ and $n_{2,k}$ in the k-th page. After *max* $\{|M_1|, |N_2|\}$ iterations, all the covering relations of P will be drawn in one of the pages because there are no covering relations between the elements of N_1 and the elements of M_1,

■

5.4 Pagenumber of Complete Multipartite Ordered Sets

An ordered set P is a *complete multipartite of length n* if the elements of P can be partitioned into n levels $L_1, L_2, \ldots, L_n$ such that all covering relations in P are only between consecutive levels and L_i, L_{i+1} is a complete bipartite ordered set for every $i < n$.

In this section, we give an approximation of the pagenumber of complete multipartite ordered sets of length 4 and 5. It is not an easy task to find the pagenumber for even a special class of complete multipartite ordered sets. Recall that $page(P) = min\{|L_1|, |L_2|\}$ for complete bipartite ordered sets and $page(P) = min\{|L_2|, |L_1|+|L_3|\}$ for tripartite ordered sets (refer to section 4.2.2 and 4.2.3). A similar approach is used in both cases to obtain optimal embedding. This approach finds a vertex cover C with a minimum size and then uses a separate page to embed all covering relations of a single element of C.

We can approximate the pagenumber of a complete multipartite ordered set P by computing its minimal vertex cover. The covering graph of P can be viewed as a bipartite graph G. We can observe that every vertex cover of P corresponds to a vertex cover in G. Efficient algorithms for finding a minimum vertex cover of bipartite graphs are available.

Notice that *if u is in a minimal vertex cover C of P, and u is in level L_i then all of L_i should be in C. Otherwise, if all of L_{i+1} and L_{i-1} are in C, then there is no need for u to be in C.* [Suppose that C does not contain all of L_{i+1}, and let *v* be in C- L_i and *w* be in C- L_{i+1}, then the edge (*v*, *w*) cannot be covered.]. Therefore a *minimal vertex cover* of *P* can be seen as a union of some levels and can be calculated polynomially.

Theorem 5.3. *Let P = (L_1, L_2, L_3, L_4) be a complete multipartite ordered set of length 4. Then, the pagenumber of P is bounded by the minimum of {$|L_1|+|L_3|$, $|L_2|+|L_4|$, $|L_2|+|L_3|-1$}.*

Proof of Theorem 5.3.

We are limiting the embeddings to those that use the same strategy for the covering relations between any two consecutive levels Li and L_{i+1} of P. Either, for every *u* in L_i, all covering relations between *u* and L_{i+1} are in the same page, or, for every *u* in L_{i+1}, all covering relations between *u* and L_i are in the same page. So, for every $i < 4$ we will use one the followings:

i. Li $\rightarrow L_{i+1}$ to represent part of the drawing indicating that for every *u* in L_i all covering relations between *u* and L_{i+1} are in the same page.

ii. $L_i \leftarrow L_{i+1}$ to represent part of the drawing indicating that for every *u* in L_{i+1} all covering relations between *u* and L_i are in the same page.

We can represent these layouts with a directed path, for example $L_1 \leftarrow L_2 \rightarrow L_3 \leftarrow L_4$ corresponds to the layout of *P* where there are $|L_2|$ pages, each containing the covering relations of a single element of L_2. These pages will cover the part $L_1 \leftarrow L_2 \rightarrow L_3$. Then, another $|L_4|$ pages is required for the part $L_3 \leftarrow L_4$ of the layout. Hence, the layout $L_1 \leftarrow L_2 \rightarrow L_3 \leftarrow L_4$ will require $|L_2| + |L_4|$ distinct pages.

It is easy to notice that there are 8 possibilities for selecting the different paths. We can, however, easily drop some of them. For example, any path that starts with $L_1 \rightarrow L_2 \rightarrow L_3$ will require at least as many pages as the one that replace $L_1 \rightarrow L_2$ by $L_1 \leftarrow L_2$. The same pages used to draw the part $L_2 \rightarrow L_3$ could also be used for the drawing of the part $L_1 \leftarrow L_2$. Similarly, we can drop those paths that end with $L_2 \leftarrow L_3 \leftarrow L_4$ since we could replace it with $L_2 \leftarrow L_3 \rightarrow L_4$. So, we are left with those embedding cases illustrated in Figure 5.8.

The possible cases of embeddings are:

1. $L_1 \rightarrow L_2 \rightarrow L_3 \rightarrow L_4$ is replaced by case 3.
2. $L_1 \rightarrow L_2 \rightarrow L_3 \leftarrow L_4$ is replaced by case 6.
3. $L_1 \rightarrow L_2 \leftarrow L_3 \rightarrow L_4$ requires $|L_1|+|L_3|$ separate pages. There is a need for $|L_1|$ pages to take care of the part $L_1 \rightarrow L_2$ and then separate $|L_3|$ pages for the part $L_2 \leftarrow L_3 \rightarrow L_4$.
4. $L_1 \rightarrow L_2 \leftarrow L_3 \leftarrow L_4$ is replaced by case 3.
5. $L_1 \leftarrow L_2 \rightarrow L_3 \rightarrow L_4$ requires $|L_2|+|L_3|$ -1 separate pages. There is a need for $|L_2|$ pages to take care of the part $L_1 \leftarrow L_2 \rightarrow L_3$ and then separate $|L_3|$ pages for the part $L_3 \rightarrow L_4$. However, in any of the pages corresponding to $L_1 \leftarrow L_2 \rightarrow L_3$ we can include all the covering relations between the top element of L_3 in the linear extension and the elements of L_4. This allows us to gain one page.
6. $L_1 \leftarrow L_2 \rightarrow L_3 \leftarrow L_4$ requires $|L_2|+|L_4|$ separate pages. The same argument applies as in the case 3.
7. $L_1 \leftarrow L_2 \leftarrow L_3 \rightarrow L_4$ requires $|L_2|+|L_3|$ -1 separate pages. The same argument applies as in the case 5.
8. $L_1 \leftarrow L_2 \leftarrow L_3 \leftarrow L_4$ is replaced by case 7.

Therefore, we show that

$$page(P) \leq min\ \{|L_1|+ |L_3|, |L_2|+|L_4|, |L_2|+|L_3|-1\}$$

■

Theorem 5.4. *Let $P = (L_1, L_2, L_3, L_4, L_5)$ be a complete multipartite ordered set of length 5. Then the pagenumber of P is bounded by*

$$min\ \{|L_2|+ |L_4|, |L_3|+ max\ (min\ (|L_1|, |L_2|-1), min\ (|L_4|-1, |L_5|))\}.$$

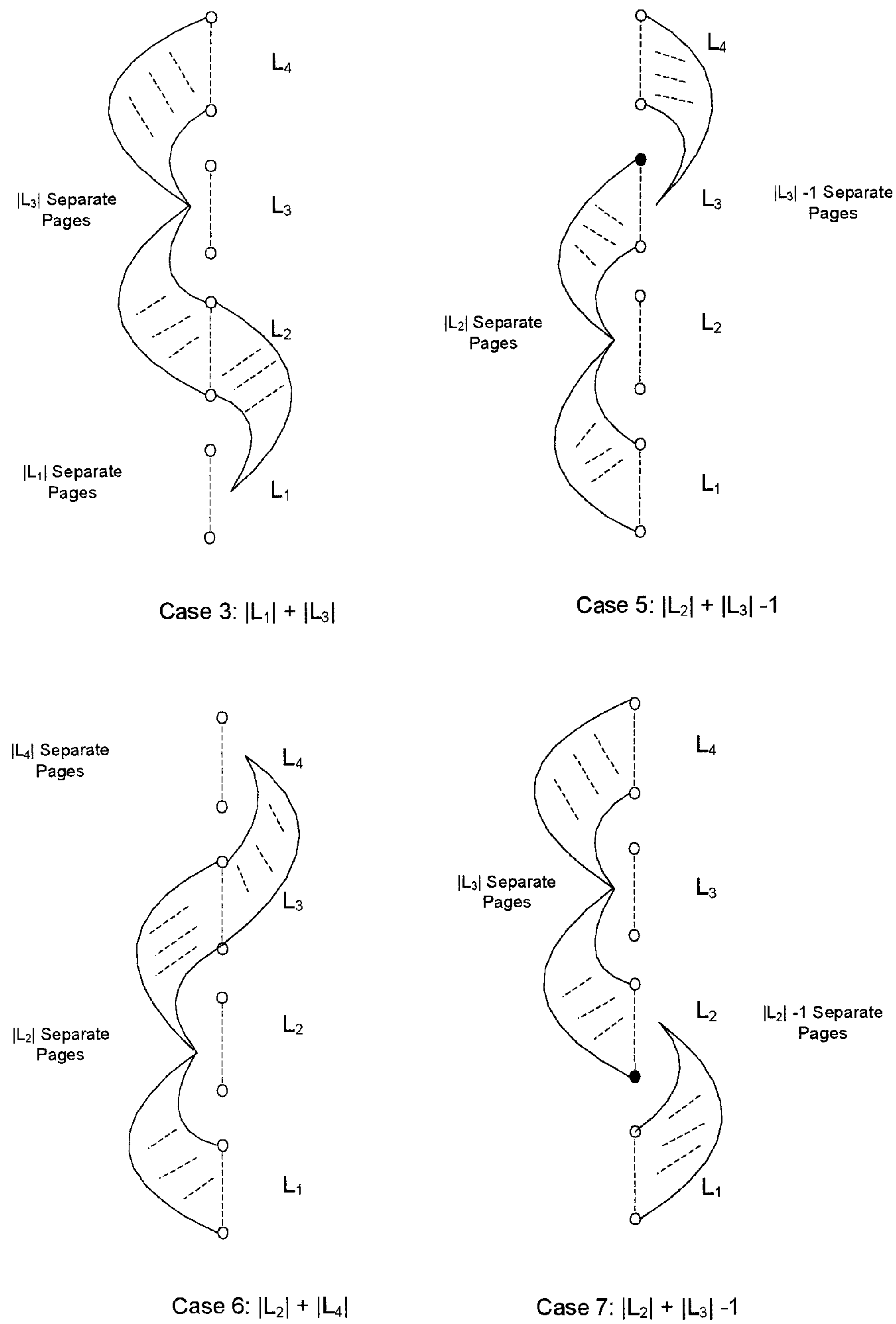


Figure 5.8: Complete multipartite ordered set of length 4 embedding cases

Proof of Theorem 5.4.

We use the same proof as in Theorem 5.3. There are 16 possibilities for selecting the different paths. We can, however, as in Theorem 5.3, drop all paths that start with $L_1 \rightarrow L_2 \rightarrow L_3$ and all paths that end with $L_3 \leftarrow L_4 \leftarrow L_5$. So, we are left with the following embedding possibilities:

1. $L_1 \leftarrow L_2 \leftarrow L_3 \rightarrow L_4 \leftarrow L_5$ requires $|L_3| + max\{|L_2| - 1, |L_5|\}$ separate pages. There is a need for $|L_3|$ pages to take care of the part $L_2 \leftarrow L_3 \rightarrow L_4$ and then $max\{|L_2| -1, |L_5|\}$ pages for the parts $L_1 \leftarrow L_2$ and L4 $\leftarrow L_5$ (Notice, in any of the pages corresponding to $L_2 \leftarrow L_3$ we can include all the covering relations between the top element of L_2 in the linear extension and the elements of L_1. This allows us to gain one page.).

2. $L_1 \rightarrow L_2 \leftarrow L_3 \rightarrow L_4 \leftarrow L_5$ requires $|L_3|+ max\{|L_1|, |L_5|\}$ separate pages. Notice that we cannot gain a page as in case 1.

3. $L_1 \leftarrow L_2 \rightarrow L_3 \rightarrow L_4 \leftarrow L_5$ is replaced by case 1 since changing $L_2 \rightarrow L_3$ by $L_2 \leftarrow L_3$ will require fewer pages

4. $L_1 \leftarrow L_2 \rightarrow L_3 \leftarrow L_4 \rightarrow L_5$ requires $|L_2|+|L_4|$ separate pages. There is a need for $|L_2|$ pages to take care of the part $L_1 \leftarrow L_2 \rightarrow L_3$ and then $|L_4|$ pages for the part $L_3 \leftarrow L_4 \rightarrow L_5$.

5. $L_1 \leftarrow L_2 \leftarrow L_3 \rightarrow L_4 \rightarrow L_5$ requires $|L_3| + max\{|L_2|-1, |L_4|-1\}$ separate pages. The same argument applies as in the case 1.

6. $L_1 \rightarrow L_2 \leftarrow L_3 \rightarrow L_4 \rightarrow L_5$ requires $|L_3|+max\{|L_1|, |L_4|-1\}$ separate pages. The same argument applies as in the case 1.

7. $L_1 \leftarrow L_2 \rightarrow L_3 \rightarrow L_4 \rightarrow L_5$ L_5 is replaced by case 5 since changing $L_2 \rightarrow L_3$ by $L_2 \leftarrow L_3$ will require fewer pages.

8. $L_1 \leftarrow L_2 \leftarrow L_3 \leftarrow L_4 \rightarrow L_5$ is replaced by case 5 since changing $L_3 \leftarrow L_4$ by $L_3 \rightarrow L_4$ will require fewer pages.

9. $L_1 \rightarrow L_2 \leftarrow L_3 \leftarrow L_4 \rightarrow L_5$ L_5 is replaced by case 6 since changing $L_3 \leftarrow L_4$ by $L_3 \rightarrow L_4$ will require fewer pages.

Therefore, we show that

$page(P) \leq min \{|L_2|+|L_4|, |L_3|+ max\ (min\ (|L_1|, |L_2|-1), min\ (|L_4|-1, |L_5|))\}.$

■

Although we have been successful to find an upper bound for the complete multipartite ordered sets of length 4 and 5, we cannot see an easy way to generate a recursive formula that can be generalized to compute the pagenumber of complete multipartite ordered set with length *n*.

Chapter 6

New Directions and Open Problems

6.1 Overview

This chapter discuses new research directions and some open problems. The research in this thesis has led to several interesting problems. Not many techniques and tools have been developed for the pagenumber problem. For instance, what is the effect of removing or inserting new elements on the pagenumber? The same question could be asked for edges. Is there a standard way to transform one optimal embedding into another optimal embedding with a specific property? What are the critical ordered sets for the pagenumber problem?

6.2 Deleting and Adding Elements in the Ordered Set

6.2.1 Deleting Elements

In designing proofs, it is interesting and helpful to know how the pagenumber changes with the removal of elements from an ordered set P. In general, it is easy to find examples where $page(P\text{-}\{x\}) < page(P)$ (see Figure 6.1) as well as examples where $page(P\text{-}\{x\}) > page(P)$ (see Figure 6.2).

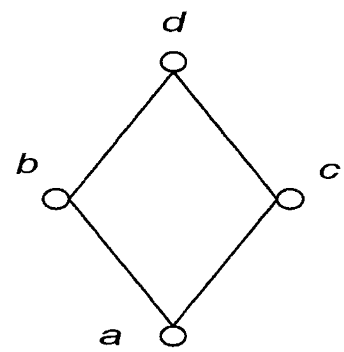


Figure 6.1: *page*(P) = 2 but *page*(P – {*x*}) = 1

The removal of irreducible elements in an ordered set is another example where we can predict the change in the pagenumber. Let *P* be an ordered set and let *x* be an element in *P*. We say that *x* is *irreducible* if *x* has a unique lower cover and a unique upper cover in *P*.

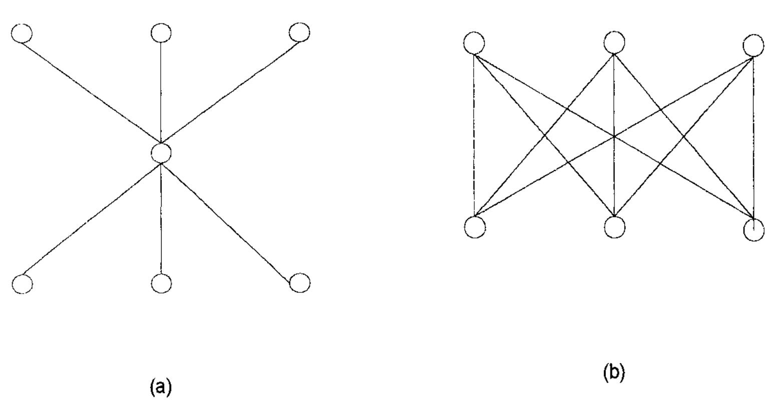


Figure 6.2: *page*(*P*) = 1 but *page*(*P* – {*x*}) = 3

Lemma 6.1 (**Alzohairi** [A96]).

Let P be an ordered set and let x be an irreducible elements such that $u \prec x \prec v$. *If* $u \prec x \prec v$ *is the unique chain from u to v then* $page(P) \leq page(P - \{x\})$.

Proof of Lemma 6.1

We can produce an upward drawing of $P - \{x\}$ from an upward drawing of P by removing x and the two edges (u, x), (x, v) and then drawing the edge (u, v). This is possible since $u \prec x \prec v$ is unique chain from u to v.

Let L' be a linear extension of $P - \{x\}$ such that $page(P - \{x\}, L') = page(P - \{x\})$. Assume that the edge (u, v) was drawn page k according to L'. It is always possible to find an horizontal line that intersects the edge (u, v) but no other edges in page k. We insert the element x at the intersection of L' and the horizontal line, and thus we can subdivide the edge (u, v) in page k into two separate edges (u, x) and (x, v). Hence, there is a linear extension L of P such that $page(P, \mathrm{L}) = page(P - \{x\}, L') = page(P - \{x\})$.

Therefore, $page(P) \leq page(P - \{x\})$.

■

Lemma 6.2.

Let P be an ordered set and let x be an irreducible element such that $u \prec x \prec v$. *If* $u \prec x \prec v$ *is not the unique chain from u to v then* $page(P - \{x\}) \leq page(P)$.

Proof of Lemma 6.2

We can obtain an upward drawing of $P - \{x\}$ from an upward drawing of P by removing x and the two edges (u, x) and (x, v). Since $u \prec x \prec v$ *is* not the unique chain from u to v, we do not need to add the edge (u, v) since $u < v$ is established by another chain.

Let L' be a linear extension of $P - \{x\}$ such that $page(P - \{x\}, L') = page(P - \{x\})$. L' can be obtained from L by removing x. Hence, L' is a linear extension of $page(P - \{x\})$.

Therefore, $page(P) \leq page(P - \{x\})$.

■

For instance, consider the ordered set P illustrated in Figure 6.1. $Page(P) = 2$ while $page(P - \{b\}) = 1$ since $a \prec b \prec c$ is not a unique chain from a to c.

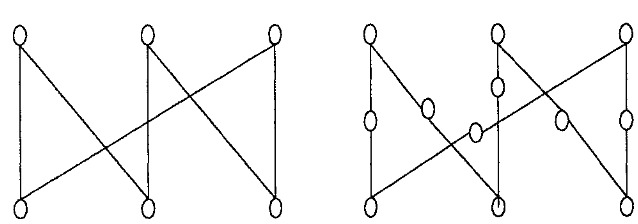
Deleting non irreducible elements could affect the planarity of the ordered set. It could also increase the pagenumber of the ordered set. For example, Figure 6.2 demonstrates that the ordered set is no longer planar after deleting a non irreducible element. The pagenumber also increased from 1 to 3 since the resulting ordered set is $P_{3,3}$.

6.2.2 Subdividing Edges

We *subdivide* a covering edge $u \prec v$ in P by adding a new element x between u and v creating new relations in P with $u \prec x$ and $x \prec v$. It is clear that edge subdivision does not change the planarity nor the non planarity of an ordered set.

Another class of ordered sets which is of interest is the class of N-free ordered sets. It is a larger class than the series parallel one and it is unknown whether we can easily compute the pagenumber for N-free ordered sets, even in the planar case.

An important fact is that every finite ordered set can be embedded as a subset in an N-free ordered set. To do that, it is enough to subdivide every covering relation in the ordered set (see Figure 6.3 for an example). Notice that there are more efficient ways to transform any ordered set into an N-free one. For instance, subdividing only the covering edges that correspond to the diagonal of an N is not enough, since new N's may be created. However, **Zaguia** [Z04] did prove that if this process is done twice then we always obtain an N-free ordered set (see Figure 6.4 for an example).



(a) (b)

Figure 6.3: N-Free Transformation.

Lemma 6.3

Let P be an ordered set with a covering relation u ≺ v. Let P' be the ordered set obtained from P by subdividing the covering relation u ≺ v. Then, page(P') ≤ page(P).

***Proof of** Lemma 6.3*

The same as the argument in the proof of Lemma 6.1

***Open Question**: Is page(P') = page(P)?*

A positive answer to this question will show that the pagenumber for N-free ordered sets will be as difficult as the pagenumber for general ordered sets

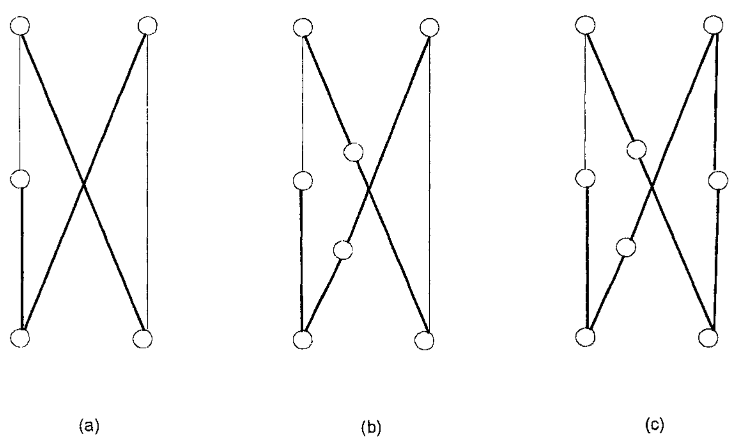


Figure 6.4: N-free transformation with two Iterations.

6.3 Transformation Techniques

In most instances of optimization problems, transformation techniques are developed to transfer one optimal solution into another optimal solution with a desired structure. The pagenumber case seems to be quite difficult since there are no known tools to use, and we have no information about the type of linear extensions that correspond to optimal page embeddings of the ordered set.

For instance, it will be interesting to have an answer to the following question: *what is the effect of switching a consecutive non comparable pair of elements in the linear extension on the pagenumber?*

For an ordered set P, it will also be very useful to *characterize the maximal elements that could be on the top of an optimal linear extension for the pagenumber.*

6.4 The Pagenumber Critical

As we mentioned in section 6.2, the removal of an element could increase the pagenumber arbitrarily. However, it is easy to see that it cannot decrease the pagenumber by more than 1.

We define an ordered set P as a *pagenumber k-critical*, or simply *k-page-critical* if *page*(P) = k and if the removal of any of its elements reduces its pagenumber, that is

$$page(P-\{x\}) < page(P) \text{ for any } x \in P.$$

We implicitly assume that a spine edge requires a page even if it is drawn on the spine. Obviously, the only page-critical ordered set P with $page(P) = 0$ is the singleton. It is also easy to check that the only page-critical ordered sets with page one is that shown in the Figure 6.5(a).

Figure 6.5(b) and Figure 6.5(c) show page-critical ordered sets with page two and three respectively. Removing any element from any of these ordered sets reduces the pagenumber of the ordered set.

A cycle C in the non-directed covering graph of P has a pagenumber 2. Moreover, the removal of any element of C will make it a tree and therefore it will lower its pagenumber. In fact, cycles are the only 2-page-critical ordered sets (see Figure 6.6).

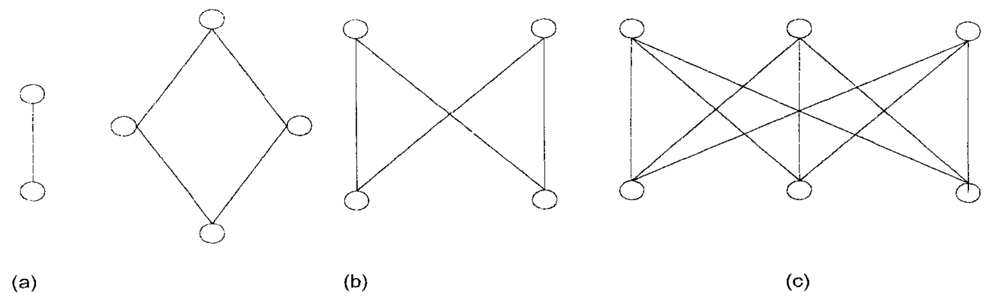


Figure 6.5: Page-critical ordered sets with page 1, 2 and 3.

The complete bipartite ordered set $K_{n,\,n}$ is an obvious example of a page-critical ordered set. Removing any element from $K_{n\,n}$ ordered set reduces the pagenumber from n to n-1 since $page(P) = min\{n, n\text{-}1\}$ for the complete bipartite ordered set. In fact and according to Theorem 5.1, the only page-critical bipartite interval orders are the complete bipartite ordered sets $K_{n,\,n}$.

Open Question: *Characterize the ordered sets which are k-page-critical.*

Removing any edge from an ordered set cannot increase the pagenumber. Perhaps the right concept for k-page-critical should be with respect to the removal of edges. For instance, the only 2-page-critical ordered sets for edges are the ordered sets where the undirected covering graph is a cycle. Also, the complete bipartite ordered set $K_{n,\,n}$ is an obvious example of a n-page-critical ordered set.

Open Question: *Characterize the ordered sets which are k-page-critical for edges.*

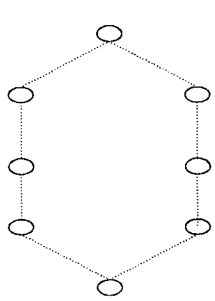


Figure 6.6: Cycle Ordered Sets

Conclusion

The importance of the visualization of complex data in today's world has become very evident. Visualization has fundamentally changed the way we present and understand large complex data sets. As a result, we can now better analyze such large data sets and arrive at better and more informed decisions.

A large number of relevant problems in different domains can be expressed as graph layout problems. The objective of the layout problems is to determine a linear layout of an input graph in such way that a certain objective cost is optimized. The book or the stack layout has multiple applications in several areas of various engineering disciplines.

In a book embedding for an *ordered set*, the vertices of the ordered set are embedded on the spine of the book to form a linear extension. The *pagenumber* is the minimum number of pages needed to draw each edge on a page such that edges drawn in the same page do not intersect. The pagenumber problem is known to be NP-complete, even if the order of elements on the spine is fixed. Therefore, we cannot expect to find a polynomial algorithm for the problem. Instead, one can either provide a heuristics approach or an algorithm that works for special class or restricted special class of ordered sets.

Along with reviews of recent literature on the pagenumber of graphs and ordered sets presented in chapter 3 and chapter 4 respectively, chapter 5 discusses several positive results achieved this thesis. We presented the first efficient algorithm with polynomial complexity for drawing the pagenumber problem of bipartite interval order. We also gave an upper bound for the pagenumber of bipartite ordered sets and the pagenumber of the complete multipartite ordered sets with length 4 and 5.

The research in this thesis has also led to several interesting problems discussed in chapter 6. The pagenumber problem seems to be very difficult since there are no known tools to use, and we have no information about the type of linear extensions that correspond to optimal page embeddings of the ordered set.

Mathematical restrictions exist on the performance of any drawing methods for graphs in general and ordered sets in particular. These limitations range from the aesthetics criteria which algorithms try to achieve to the computational resources required by the algorithms. In many cases, there are also trade-offs between various aesthetic criteria and the computational resources. To come up with effective drawing techniques, an appreciation of theses limits and trade-offs are required.

Bibliography

[A96] M. Alzohairi, "The Pagenumber of Ordered Sets," *PhD Thesis*, University of Ottawa, 1996.

[AR96] M. Alzohairi and I. Rival, "Series-Parallel Planar Ordered Sets have Pagenumber Two," *Graph Drawing*, pp. 11-24, 1996.

[ARK01] M. Alzohairi, I. Rival and A. Kostochka, "The Pagenumber of Spherical Lattices is Unbounded," Arab J. Math. Sci., vol. 7, no. 1, pp. 79-82, 2001.

[BCLR96] S. Bhatt, F. Chung, F. Leighton and A. L. Rosenberg, "Scheduling Tree-Dags using FIFO Queues: a Control-Memory Trade-off," *J. Parallel Distrib. Comput.*, vol. 33, pp. 55-68, 1996.

[BETT99] G. Battista, P. Eades, R. Tamassia and I. Tollis. *Graph Drawing: Algorithms for the Visualization of Graphs,* Prentice Hall, 1999.

[BK79] F. Bernhart and P. Kainen, "The Book Thickness of a Graph," *J. Combin. Theory Ser. B*, vol. 27, pp. 320-331, 1979.

[BS84] L. Buss and P. Shot, "On the Pagenumber of Planar Graphs," *Prec. 16th Ann. ACM Syrup. on Theory of Computing*, pp. 98-100,1984.

[BSWW99] T. Biedl, T. Shermer, S. Whitesides and S. Wismath, "Bounds for Orthogonal 3-D Graph Drawing," *J. Graph Algorithms Appl.*, vol. 3, no. 4, pp. 63–79, 1999.

[CLR87] F. Chung, F. Leighton and A. Rosenberg, "Embedding Graphs in Books: a Layout Problem with Applications to VSLI Design," *SIAM Journal on Algebraic and Discrete Methods*, vol. 8, pp.33-58, 1987.

[DP90] B. Davey and H. Priestley, *Introduction to Lattices and Order*, Cambridge University Press, Cambridge, 1990.

[DPS02] J. D´iaz, J. Petit and M. Serna. A Survey of Graph Layout Problems. *ACM Computing Surveys,* vol. 34, no 3, pp. 313-356, 2002.

[DW04] V. Dujmović and D. Wood, "On Linear Layouts of Graphs," *Discrete Mathematics and Theoretical Computer Science*, vol. 6, no. 2, pp. 339-358, 2004.

[E97] T. Enod, "The Pagenumber of Toroidal Graph is at Most Seven," *Discrete Math.*, vol. 175, no. 1-3, pp. 87-96, 1997.

[EI71] S. Even and A. Itai, "Queues, Stacks and Graphs," *Theory of Machines and Computations,* Academic Press, New York, 71-86, 1971.

[ENO97] H. Enomoto, T. Nakamigawa and K. Ota, "On the Pagenumber of Complete Bipartite Graphs," *Journal of Combinatorial Theory Series B*, vol. 71, no. 1, pp. 111–120, 1997.

[F92] S. Felsner, "Interval Orders: Combinatorial Structure and Algorithms," *PhD Thesis*, Technische Universität Berlin, 1992.

[FMP95] H. Fraysseix, P. Mendez and J. Pach, "A Left Search Algorithm for Planar Graphs," *Discrete Comput. Geom.*, vol. 13, pp. 459-468, 1995.

[G83] P. Grillet, "Maximal Chains and Antichains," *Fund. Meth.*, vol.15, pp. 157-167, 1983.

[G95] J. Ganley, "Stack and Queue Layouts of Halin Graphs," 1995, manuscript.

[GDLW02] E. Giacomo, W. Didimo, G. Liotta and S. Wismath, "Book Embeddings and Point-Set Embeddings of Series-Parallel Digraphs," *Graph Drawing*, pp. 162-173, 2002.

[GEC98] N. Gershon, S. Eick and S. Card, "Information Visualization," *Interactions*, vol. 5, no. 2, pp. 9-15, 1998.

[GH01] J. Ganley and L. Heath, "The Pagenumber of *k*-trees is *O*(*k*)," *Discrete Appl. Math.*, vol. 109, no. 3, pp. 215-221, 2001.

[GH99] C. Gavoille and N. Hanusse, "Compact Routing Tables For Graphs Of Bounded Genus," In J. Wiedermann, P. Van Emde Boas, and M. Nielsen, Eds., *Proc. 26th International Colloquium on Automata, Languages and Programming (ICALP '99)*, vol. 1644 of *Lecture Notes in Computer Science*, pp. 351-360, Springer, 1999.

[GJMP80] M. Garey, D. Johnson, G. Miller and C. Papadimitriou, "The Complexity of Coloring Circular Arcs and Chords," *SIAM J. Alg. Dis. Meth.*, vol. 1, pp. 185-200, 1980.

[GKS89] Z. Galil, R. Kannan and E. Szemerédi, "On Nontrivial Separators for *k*-page Graphs and Simulations by Nondeterministic One-tape Turing Machines," *J. Comput. System Sci.*, vol. 38, no. 1, pp. 134-149, 1989.

[GM04] E. Giacomo and H. Meijer, "Track Drawings of Graphs with Constant Queue Number," In G. LIOTTA, ed., *Proc. 11th International Symp. on Graph Drawing 2003*, vol. 2912 of *Lecture Notes in Comput. Sci.*, pp. 214-225, Springer, 2004.

[H84] L. Heath, "Embedding Planar Graphs in Seven Pages," Prec. 25th Ann. Symp. on Foundations of Computer Science, pp. 74-83, 1984.

[H93] L. Hung, "A Planar Poset which Requires Four Pages," *Ars Combin.*, vol. 35, pp. 291-302, 1993.

[HI92] L. Heath and S. Istrail, "The Pagenumber of Genus g Graphs is O(g)," *J. Assc. Comput. Mach.*, vol. 39, no. 3, pp. 479-501, 1992.

[HP97] L. Heath and S. Pemmaraju, "Stack and Queue Layouts of POSETS," SIAM Journal on Discrete Mathematics, vol. 10, no. 4, pp. 599-625, 1997.

[HP99] L. Heath and S. Pemmaraju, "Stack and Queue Layouts of Directed Acyclic Graphs: Part II," *SIAM Journal on Computing*, vol. 28, pp. 1588-1626, 1999.

[HPT99] L. Heath, S. Pemmaraju and A. Trenk, "Stack and Queue Layouts of Directed Acyclic Graphs: Part I," *SIAM Journal on Computing*, vol.28, pp. 1510-1539, 1999.

[HS97] T. Hasunuma and Y. Shibata, "Embedding de Bruijn, Kautz and Shuffle-Exchange Networks in Books," *Discrete Applied Mathematics*, vol. 78, no. 1-3, pp. 103-116, 1997.

[I86] S. Istrail. "An Algorithm for Embedding Planar Graphs in Six Pages," manuscript, 1986.

[IOS96] I. Rival, "Introducing Ordered Sets," Lectures on Ordered Sets, http://www.site.uottawa.ca/dept/algorithms/order, 1996.

[K77] D. Kelly, "The 3-irreducible Partially Ordered Sets," *Can. J. Math.*, vol. 29, pp. 367-383, 1977.

[KRSZ02] N. Kapoor, M. Russell, I. Stojmenovic and A. Zomaya, "A Genetic Algorithm for Finding the Pagenumber of Interconnection Networks", *Journal of Parallel and Distributed Computing*, vol. 62, no. 2, pp. 267-283, 2002.

[M94a] S. Maltz, "Graphs with E Edges have Pagenumber O($\sqrt{E}$)," *J. Algorithms*, vol. 17, no. 1, pp. 71-84, 1994.

[M94b] S. Maltz, "Genus g Graphs Have Pagenumber O($\sqrt{g}$)," *Journal of Algorithms*, vol. 17, no. 1, pp. 85-10, 1994.

[MOS98] P. Mutzel, T. Odenthal and M. Scharbrodt, "The Thickness of Graphs: a Survey," *Graphs Combin.*, vol.14, no. 1, pp. 59-73, 1998.

[MR01] J. Munro and V. Raman, "Succinct Representation of Balanced Parentheses and Static Trees," *SIAM Journal on Computing*, vol. 31, no.

3, pp. 762-776, 2001.

[MWW88] D. Muder, M. Weaver and D. West, "Pagenumber of Complete Bipartite graphs," *Journal of Graph Theory*, vol. 12, pp. 469-489, 1988.

[NP89] R. Nowakowski and A. Parker, "Ordered Sets, Pagenumbers, and Planarity," *Order*, vol. 6, pp. 209-218, 1989.

[O91] B. Obrenic, "Embedding de Bruijn and Shuffle-Exchange Graphs in Five Pages (preliminary version)," *ACM Symposium On Parallel Algorithms And Architectures, Proceedings Of The Third Annual ACM symposium on Parallel algorithms and Architectures,* Hilton Head, South Carolina, United States, pp. 137 -146, 1991.

[P92] S. Pemmaraju, "Exploring the Powers of Stacks and Queues via Graph Layouts," Ph.D. Thesis, Virginia Polytechnic Institute and State University, U.S.A., 1992.

[R83a] I. Rival, "Optimal Linear Extension by Interchanging Chains," *Prc. Amer. Math Soc.*, vol. 89, pp. 387-394, 1983.

[R83b] A. Rosenberg, "The Diogenes Approach to Testable Fault-Tolerant VLSI Processor Arrays," *IEEE Trans. Comput.*, vol. C-32, pp. 902-910, 1983.

[R86] I. Rival, "Stories about Order and the Letter N (en)," Combinatorics and Ordered Sets (Arcata, Calif., 1985), Contemp. Math., vol. 57, pp. 263-285, Amer. Math. Soc., Providence, RI, 1986.

[S89] M. Syslo, "Bounds to the Page Number of Partially Ordered Sets," *In Proc. of the 15th Intl. Workshop on Graph-Theoretic Concepts in Comput. Sc.*, M. Nagl, ed., Springer-Verlag, Berlin, pp. 181–195, 1989.

[VTL82] J. Valdes, R. Tarjan, E. Lawler. The Recognition of Series-Parallel Digraphs. *SAM J. Computing*, 11:298-313, 1982.

[W98] R. Webber. Finding the Best Viewpoint for Three-Dimensional Graph Drawings. PHD Thesis, The University of Newcastle, 1998.

[T72] R. Tarjan, "Sorting using Networks of Queues and Stacks," *J. Assoc. Comput. Mach.*, vol. 19, pp. 341-346, 1972.

[Y86] M Yannakakis, "Four Pages are Necessary and Sufficient for Planar Graphs," *Proceedings of the eighteenth annual ACM symposium on Theory of computing*, ACM Press, pp. 104-108, 1986.

[Y89] M. Yannakais, "Embedding Planar Graphs in Four Pages," *Journal Computer and System Science*, vol. 38, no. 1, pp. 36-67, 1989.

[Z04] N. Zaguia. *Private Communication*, 2004.