An Efficient Heuristic for Crossing Minimisation and its Applications

A thesis submitted for the degree of
Doctor of Philosophy

Alex Newcombe
B.Sc. (Hons)

School of Computer Science,
Engineering & Mathematics

Faculty of Science & Engineering

Flinders University

November 2019

Contents

Contents i

List of Tables v

List of Figures viii

Summary xii

Declaration xvi

Publications xvii

Acknowledgements xviii

Notation xix

1 Introduction 1

1.1 Definitions and concepts 2

1.1.1 Basic definitions 3

1.1.2 Graph products 6

1.1.3 Asymptotic notation and complexity 6

1.1.4 Embeddings, drawings and crossing numbers 8

1.2 Crossing numbers 12

1.2.1 Variants of the crossing number 15

1.3 Motivating example 16

2 A new crossing minimisation heuristic based on star insertion 20

2.1 Related work 20

2.1.1 Exact algorithms 21

2.1.2 Heuristic methods 25

2.2 Proposed heuristic method 31

2.3 Design methodology 35

2.3.1 Initial embedding schemes 35

2.3.2 Minimisation schemes 39

2.3.3 Efficiently handling the dual graphs 40

2.3.4 Pre- and post-processing schemes 42

2.3.5 Data structures 43

2.3.6 Difficulties arising from fixed embeddings 45

2.3.7 Updating the crossing lists 49

2.4 Implementation and Runtime 57

2.4.1 Implementation 58

2.4.2 Runtime 60

2.5 Experiments 62

2.5.1 Experimental setup 62

2.5.2 KnownCR graphs 67

2.5.3 Rome graphs 74

2.5.4 AT&T graphs 75

2.5.5 ISCA graphs 78

2.5.6 Complete graphs 80

2.5.7 Complete bipartite graphs 81

2.5.8 Concluding observations 83

2.6 Incremental post-processing 86

3 New exact results relating to crossing numbers 91

3.1 Minimal cubic graphs with crossing number at least k 93

3.2 Cartesian product of a Sunlet graph and a Star graph 103

3.2.1 Upper Bound 105

3.2.2 Exact results 107

3.3 Crossing numbers of graph products involving one small graph 113

3.3.1 Cartesian products involving one small graph 118

3.3.2 Join products involving one small graph 127

4 New bounds and conjectures relating to crossing numbers 138

4.1 Cartesian products involving cycles 140

4.2 Join product of a complete graph and a discrete graph 154

4.3 Generalised Petersen graphs 162

4.4 n-dimensional hypercube graphs 173

4.5 Sheehan's maximal uniquely Hamiltonian graphs 176

5 Conclusions and future work 179

5.1 Future work arising from Chapter 2 181

5.2 Future work arising from Chapter 3 182

5.3 Future work arising from Chapter 4 183

Appendix A Counting crossings in a drawing 186

Appendix B Proof of Theorem 4.19 **189**

Appendix C Predicted crossing numbers of graphs resulting from products **198**

Bibliography **211**

List of Tables

2.16 Experimental results on the KnownCR graphs 68

2.25 Quickcross initial embeddings on the KnownCR graphs 73

2.27 Experimental results on the Rome graphs - 100 permutations . 76

2.28 Experimental results on the Rome graphs - 500 permutations . 76

2.29 Experimental results on the AT&T graphs - 100 permutations 77

2.30 Experimental results on the AT&T graphs - 500 permutations 78

2.31 Experimental results on the ISCA graphs. 79

2.32 Runtimes for the ISCA graphs. 80

2.33 Experimental results on the complete graphs. 82

2.34 Runtimes for the complete graphs. 83

2.35 Experimental results on the complete bipartite graphs. 84

2.36 Runtimes for the complete bipartite graphs. 85

2.37 Q-inc results on the KnownCR graphs. 88

2.38 Q-inc results on the Rome graphs. 89

2.39 Q-inc results on the AT&T graphs. 89

2.40 Q-inc results on the ISCA graphs. 90

3.1 Minimal cubic graphs with crossing number at least k 94

3.3 Maximum crossing number of a cubic graph of order n 96

3.8 Quickcross results for Cartesian product of a sunlet graph and a star . . . 105

3.16 Known crossing numbers of Cartesian products of graphs on six vertices with paths . . . 118

3.19 Known crossing numbers of Cartesian products of graphs on six vertices with cycles . . . 121

3.26 Crossing number of Cartesian products of six vertex graphs with small cycles . . . 124

3.27 Known crossing numbers of Cartesian products of graphs on six vertices with stars . . . 125

3.30 Known crossing numbers of join products of graphs on five vertices with discrete graphs, paths and cycles . . . 129

3.34 Known crossing numbers of join products of graphs on six vertices with discrete graphs, paths and cycles . . . 132

3.35 Continuation of Table 3.34. . . . 133

4.6 Known crossing numbers of $G\Box C_m$ for small graphs . . . 149

4.9 Crossing numbers of $\widehat{G}$ for small graphs . . . 153

4.13 The upper bound in Theorem 4.20 when n is fixed . . . 160

4.14 The upper bound in Theorem 4.20 when d is fixed . . . 161

4.16 Known crossing numbers of $GP(n, 4)$ . . . 164

4.17 Minimum found number of crossings by Quickcross for the graphs $GP(n, 4)$. . . . 165

4.18 Known crossing numbers of $GP(n, 5)$ . . . 165

4.20 Minimum found number of crossings by Quickcross for the graphs $GP(n, 5)$. . . . 166

4.33 n-dimensional cube experimental results . . . 175

4.35 Sheehan graph experimental results 178

5.1 Ranking of importance for unknown results related to graph products. 184

B.1 $f(n, d)$ for different parities of n and d. 190

C.1 Predicted crossing numbers for the Cartesian product of a five vertex graph with cycles and stars. 200

C.2 Predicted crossing numbers for the join product of a five vertex graph with discrete graphs, paths and cycles 201

C.3 Predicted crossing numbers for the Cartesian product of a six vertex graph with paths, cycles and stars. 202

C.4 Predicted crossing numbers for the join product of a six vertex graph with discrete graphs, paths and cycles 206

List of Figures

1.1 Two drawings of $K_{3,3}$. 1

1.2 Planarisation of a drawing 10

1.3 Drawing modifications which remove crossings 12

1.6 Two drawings of the Petersen graph 18

2.1 Star insertion path example 34

2.2 Contracting a cycle in the dual graph 36

2.3 Edge contractions in the dual graph 41

2.4 Crossing orientation with respect to an edge 44

2.5 Data structures example 44

2.6 A pair of edges crossing twice example 46

2.7 Modification to rectify edges crossing twice 47

2.9 Incident edges which cross example 48

2.10 Modification to rectify incident edge crossings 48

2.11 Updating crossing order lists example 1 51

2.12 Updating crossing order lists example 2 52

2.13 Constructing crossing order lists example 54

2.14 Identifying new crossing orientations 57

2.17 Quickcross heuristic runtime plot for the graphs $G_i \Box P_j$ 69

2.18 Quickcross heuristic runtime plot for the graphs $G_i \Box C_j$ 70

2.19 Quickcross heuristic runtime plot for the graphs $C_i \Box C_j$ 70

2.20 Quickcross heuristic runtime plot for the graphs $P(j, 3)$ 71

2.21 OGDF heuristic runtime plot for the graphs $G_i \Box P_j$ 71

2.22 OGDF heuristic runtime plot for the graphs $G_i \Box C_j$ 72

2.23 OGDF heuristic runtime plot for the graphs $C_i \Box C_j$ 72

2.24 OGDF heuristic runtime plot for the graphs $P(j, 3)$ 73

2.26 Quickcross initial embedding runtime - KnownCR graphs . . . 74

3.2 The four smallest cubic graphs with crossing number 7 95

3.4 McGee graph plus an edge drawn with 10 crossings 99

3.5 McGee graph plus an edge drawn with 9 crossings 99

3.6 A cubic graph on 28 vertices with crossing number 10 101

3.7 The Sunlet graph, $\mathcal{S}_6$. 105

3.9 Planar drawing of $\mathcal{S}_6 \Box K_{1,1}$. 106

3.10 Construction of drawings of $K_{1,m} \Box C_n$ 107

3.11 The subgraph H_i of $\mathcal{S}_n \Box K_{1,3}$ 109

3.12 The subgraph H_i' of $\mathcal{S}_n \Box K_{1,2}$ 109

3.13 The drawing of F_i, if F_i is not crossed by itself 111

3.14 A list of all 21 connected graphs on five vertices 116

3.15 A list of all 156 graphs on six vertices 117

3.17 $G^6_{110} \Box P_n$ drawn with $3n - 1$ crossings 119

3.18 $G^6_{137} \Box P_n$ drawn with $4n$ crossings 120

3.20 $G^6_{63} \Box C_n$ drawn with $2n$ crossings 122

3.21 $G^6_{64} \Box C_n$ drawn with $2n$ crossings 122

3.22 $G^6_{98} \Box C_n$ drawn with $3n$ crossings 123

3.23 $G^6_{75} \Box C_n$ drawn with $2n$ crossings 123

3.24 $G^6_{77}\Box C_n$ drawn with $2n$ crossings 123

3.25 $G^6_{92}\Box C_n$ drawn with $3n$ crossings 124

3.28 $G^6_{62}\Box S_n$ drawn with $5\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$ crossings 126

3.29 $G^6_{121}\Box S_n$ drawn with $6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n$ crossings 127

3.31 $G^5_9+D_n$ drawn with $n(n-1)$ crossings 130

3.32 $G^5_8+D_n$ drawn with $Z(5,n)$ crossings 131

3.33 $G^5_{17}+C_n$ drawn with $Z(5,n)+n+3$ crossings 131

3.36 $G^6_{104}+D_n$ drawn with $Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor$ crossings 134

3.37 $G^6_{140}+D_n$ drawn with $Z(6,n)+2\left\lfloor\frac{3n}{2}\right\rfloor$ crossings 134

3.38 $G^6_{85}+P_n$ drawn with $Z(6,n)+n+1$ crossings 135

3.39 $G^6_{79}+C_n$ drawn with $Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor+3$ crossings 136

3.40 $G^6_{85}+C_n$ drawn with $Z(6,n)+n+3$ crossings 136

4.1 An example of G and $\widehat{G}$ 141

4.2 The cyclic ordering of edges around v_1 in $\widehat{D}'$. 143

4.3 Modifications to $\widehat{D}'$. 144

4.4 Layout of D_i in a circle. 146

4.5 Modifications to produce a drawing of $G\Box C_m$. 147

4.7 Drawings of $\widehat{P}_m$ and $K_{1,1,1,m}$. 150

4.8 Modifications to $\widehat{C}_m$. 151

4.10 A drawing of K_3+D_3. 154

4.11 Cylinder drawings of K_7 and K_8 158

4.12 Spiral cylinder drawings of K_3+D_4 and K_4+D_4 159

4.15 Drawings of $GP(6,2)$ and $GP(9,4)$ 162

4.19 $GP(17,5)$ drawn with 13 crossings 166

4.21 $GP(25,5)$ drawn with 18 crossings 169

4.22 $GP(5h, 5)$ drawn with $4h$ crossings 170

4.24 $GP(5h + 1, 5)$ drawn with $4h + 2$ crossings 170

4.26 $GP(5h + 2, 5)$ drawn with $4h + 4$ crossings 171

4.28 $GP(5h + 3, 5)$ drawn with $4h + 4$ crossings 172

4.30 $GP(5h + 4, 5)$ drawn with $4h + 4$ crossings 172

4.32 The 4-dimensional hypercube graph. 174

4.34 Sheehan graph on 16 vertices. 176

B.2 Vertex and index layouts for drawings of $K_n + D_d$ 191

B.3 Spiral cylinder drawings of $K_3 + D_4$ and $K_4 + D_4$ 191

B.4 The six types of crossings which are removed from K_{n+d} . . . 193

Summary

This thesis aims to provide a broad study of the crossing number problem. In particular, we describe a new heuristic for crossing minimisation, and demonstrate that it performs comparably to the state of the art existing heuristics. We then take advantage of the new heuristic to help in developing various new results and conjectures related to crossing numbers. We begin by introducing crossing numbers, along with the required graph theory concepts, in Chapter 1.

Then, in Chapter 2, we introduce a new heuristic method for minimising crossings in graphs. Our basic approach is to begin with a, presumably sub-optimal, drawing of a graph and then use solutions to the related star insertion problem to iteratively obtain new drawings with fewer crossings. We implement the heuristic, dubbed Quickcross, and spend a significant amount of effort making the implementation efficient for practical use. To this end, in Section 2.3, we discuss several of the schemes and design features of the implementation. Then, in Section 2.5, we make experimental comparisons between the various combinations of schemes of Quickcross by running it on several sets of graphs which have previously been used for benchmarking purpose. Where appropriate, we also make comparisons with the current state-of-the-art crossing minimisation heuristics included in the OGDF package. We observe that Quickcross compares well in many instances, and identify several consistently strong performing schemes of Quickcross which we recommend for practical use.

In Chapter 3, we consider several problems relating to crossing numbers, and develop some new, exact results. Even though Quickcross is only a heuristic, for the problems considered in Chapter 3 it is often a useful tool in developing these results. First, in Section 3.1, we consider the problem of determining the minimum number of vertices of a cubic graph which has crossing number at least k. By using Quickcross to perform a significant body of computations, we are able to extend the previously known results from $k = 8$ up to $k = 11$. A corollary of these results confirms a folklore belief that the Coxeter graph is an example of a minimal cubic graph with crossing number at least 11. Next, in Section 3.2, we consider the crossing number of the Cartesian product of a sunlet graph and a star. We use Quickcross to predict what the crossing number of this family of graphs might be, and then confirm our prediction for the first few cases where the size of the star is fixed. We also determine general upper bounds and conjecture that these are tight. Then, in Section 3.3, we consider all families of graphs resulting from the graph product of a fixed small graph with an arbitrarily large path, cycle, star or discrete graph. There has been considerable effort over the last three decades to determine the crossing numbers of such families of graphs. We expand upon the known results by determining the crossing number for 29 new such families. We then use Quickcross to predict formulas for the crossing numbers of all remaining families of such graphs where the crossing numbers are unknown. After the results determined in Chapter 3 are taken into account, there are still 609 results of this type remaining to be determined.

In Chapter 4, we consider several problems relating to crossing numbers, and develop some new bounds and conjectures. For these problems, Quickcross serves more as a guide our investigations rather than a direct aid. In Section 4.1, we build upon an observation made during our research in Section 3.3 concerning the Cartesian product of any fixed graph with an

arbitrarily large cycle. We propose that for any graph, the crossing number of its Cartesian product with an appropriately large cycle obeys a simple formula related to the crossing number of a, much smaller, auxiliary graph. We also demonstrate that every known result of this type in literature agrees with our proposal. Then, in Section 4.2, we consider the join product of a complete graph and a discrete graph. The crossing number of the first few small cases of this family has already been determined. For the general case, we determine recursive lower bounds and then also determine new upper bounds with a drawing procedure. Our formula for the upper bound has a simple form and a pleasing interpretation. Next, in Section 4.3, we consider the generalised Petersen graphs. We discuss the somewhat scattered history of results for these graphs and then experimentally examine the smallest infinite family of these graphs for which no genuine investigation has been conducted to date. We determine drawing procedures for these graphs, and so provide a new upper bound for the crossing number. We also provide evidence that this bound is tight. Next, in Section 4.4, we consider the n-cube graphs. The number of vertices in the n-cube grows exponentially with n and as such, these graphs are only tractable with heuristic methods for around $n \leq 10$. During our experimentation, we obtained drawings of the n-cube for $7 \leq n \leq 8$ possessing strictly fewer crossings than a previously thought tight upper bound. Although we were not the first researchers to notice that the upper bound was not tight, this is the first time a heuristic method has successfully found such drawings. Lastly, in Section 4.5, we consider the Sheehan graphs, which are a family of graphs that are of interest in a different graph theory context; specifically, the Hamiltonian cycle problem. The crossing number of these graphs has not previously been considered and so we aim to make some preliminary observations. We observe that the tested crossing minimisation heuristics have significant difficulty finding solutions with few crossings. After a significant amount of effort, we obtain drawings for which the numbers of crossings is very close to a particular formula.

Finally, we conclude this thesis in Chapter 5 by summarising the results obtained throughout, and discuss some of the ripe areas of future research. This includes both a discussion of new heuristic approaches that might prove fruitful, and also ideas on how best to further extend the theoretical results developed in this thesis.

Declaration

I certify that this thesis does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any university; and that to the best of my knowledge and belief it does not contain any material previously published or written by another person except where due reference is made in the text.

Alex Newcombe

Publications

The following list details research within this thesis that is either published or currently submitted for review.

Chapter 2:

- This chapter describes joint work with Dr. M. Haythorpe and Dr. K. Clancy. A significantly condensed version of the chapter was published in the Journal of Graph Algorithms and Applications volume 23(2) in 2019 [40].

Chapter 3:

- Section 3.1 describes joint work with Dr. M. Haythorpe, Dr. K. Clancy and Dr. E. Pegg Jr. A condensed version of this section is submitted to the Journal of Graph Theory.

- Section 3.2 describes joint work with Dr. M. Haythorpe. Part of this section is included in a manuscript, which was published in the Bulletin of the Australian Mathematical Society volume 100(1) in 2019 [78].

- Section 3.3 describes joint work with Dr. M. Haythorpe and Dr. K. Clancy. Parts of this section appear in a manuscript submitted to the Australasian Journal of Combinatorics as well as a different manuscript submitted to the Journal of Combinatorial Mathematics and Combinatorial Computing.

Acknowledgements

It is an honour to be able to thank the people, whose support, made this thesis possible.

First, I thank my principal supervisor Dr. Michael Haythorpe for his unending perseverance, his expertise and for the enjoyable time that we spend researching together. I thank my associate supervisors Prof. Jerzy Filar and A/Prof. Vladimir Ejov for their strong leadership and their continual support over the last four years.

I thank Dr. Nurulla Azamov, A/Prof. Murk Bottema and A/Prof. Alan Branford who all introduced me to true mathematics. Thank you to Dr. Gobert Lee for introducing me to the joys of teaching. I am grateful to my friends, new and old, from Flinders University who helped me in one way or another, and often, significantly more than they know. In no particular order, they include:

Asghar, Kieran, Tom, Pouya, David, Yang, Bree, Serguei, Dafiana, Iwan, Hayden, Mariusz.

I thank the Australian Government Research Training Program for being given the opportunity and support to pursue this degree.

Finally, I give my deepest gratitude to my family for always believing in me and always being nearby, even if we go long periods of time without seeing each other. My extended family in Queensland for accepting me. My fiancé Deborah for her love, support and company on the late work nights.

Notation

The following notations are used throughout the thesis.

$cr(G)$	The crossing number of G
$cr_D(G)$	The number of crossings in the drawing D of G
C_n	The cycle graph with n vertices
$d(v)$	The degree of a vertex $v \in V(G)$
D_G	A drawing of G into the plane
D_n	The discrete (empty) graph with n vertices
$\Delta(G)$	The maximum degree of any vertex in G
$E(G)$	The set of edges of G
G	A graph with vertex set $V(G)$ and edge set $E(G)$
G_i^5	One of the 21 non-isomorphic, connected graphs on 5 vertices where i is given in Figure 3.14
G_i^6	One of the 156 non-isomorphic graphs on 6 vertices where i is given in Figure 3.15
$H(n)$	The conjectured value of $cr(K_n)$, $\frac{1}{4}\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor\left\lfloor\frac{n-2}{2}\right\rfloor\left\lfloor\frac{n-3}{2}\right\rfloor$
$G \square H$	The Cartesian product of graphs G and H
$G \cong H$	The graphs G and H are isomorphic
$G + H$	The join product of graphs G and H
K_n	The complete graph with n vertices
$K_{n_1,n_2,\ldots,n_k}$	The complete multipartite graph with partitions of cardinalities $n_1, n_2, \ldots, n_k$

$N_G(v)$	The set of vertices which are adjacent to vertex v
$O(f(x))$	A function bounded from above by a constant positive multiple of $f(x)$ for all $x > N$ for some N
$\Omega(f(x))$	A function bounded from below by a constant positive multiple of $f(x)$ for all $x > N$ for some N
Π	A combinatorial embedding of a planar graph
Π^*	Dual graph of the embedding Π
P_n	The path graph with $n+1$ vertices
S_n	The star graph with $n+1$ vertices (also $K_{n,1}$)
$\mathcal{S}_n$	The sunlet graph with $2n$ vertices
$\Theta(f(x))$	A function bounded both above and below by a constant positive multiple of $f(x)$ for all $x > N$ for some N
(u, v)	The edge with end vertices u and v
$V(G)$	The set of vertices of G
$Z(n_1, n_2)$	The conjectured value of $cr(K_{n_1,n_2})$, $\left\lfloor \frac{n_1}{2} \right\rfloor \left\lfloor \frac{n_1-1}{2} \right\rfloor \left\lfloor \frac{n_2}{2} \right\rfloor \left\lfloor \frac{n_2-1}{2} \right\rfloor$

Chapter 1

Introduction

In this thesis, we are primarily concerned with drawings of graphs. Consider the complete bipartite graph $K_{3,3}$, as displayed in its typical drawing in Figure 1.1 (a). A natural question to ask is: *what is the best way to draw this graph?* One possibility is to draw $K_{3,3}$ with the aim of making the number of edges which cross each other small. Then, after some thought, a drawing of $K_{3,3}$ with only a single edge crossing may be devised, such as in Figure 1.1 (b). If it can be proved that we have drawn $K_{3,3}$ with the minimum possible number of edge crossings, then this number is the *crossing number* of $K_{3,3}$. Informally, the crossing number problem asks: *Given a graph G, what is the minimal number of edge crossings in any drawing of G.* The act of trying to find drawings with few crossings is referred to as *crossing minimisation.*

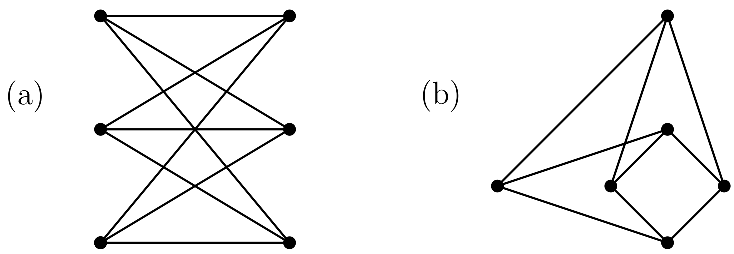


Figure 1.1: Two drawings of $K_{3,3}$.

Crossing minimisation was noticed by Paul Turán in 1944 and described in his welcome note to the first issue of the distinguished Journal of Graph

Theory [137]. The observation of Turán specifically concerned crossing minimisation in the family of complete bipartite graphs and this problem is now commonly referred to as Turán's Brick Factory Problem. The first attempts at solutions were published independently by Kazimierz Zarankiewicz and Kazimierz Urbanik in [147] and [138]. However, over 10 years later, their arguments were found to be flawed and despite significant effort, the problem remains open today [71]. Nonetheless, their proposed result is believed to be correct, and the general conjecture has become known as Zarankiewicz' conjecture.

The crossing number of the complete graphs is perhaps a more natural question to ask compared to the complete bipartite graphs. The first serious investigations into the complete graphs were made by Anthony Hill in 1958 and the conjectured result, which also remains open, has been named after Hill [68]. The general difficulty of studying crossing numbers is perhaps best illustrated by these two 'original' conjectures, which have withstood ongoing attacks at solutions for over half a century despite significant and ongoing effort [113, 108, 4, 45].

In this introductory chapter, we begin by providing some concepts and definitions from graph theory. Then, we are in the position to review some of the landmark results related to crossing numbers and also some of the variants of crossing numbers. Later, in each chapter of this thesis, we provide a focused literature review specifically on results related to the content in that chapter.

1.1 Definitions and concepts

Throughout this thesis we shall be working with many different concepts and as such require many definitions. Here we will define some general notions that will be used throughout the thesis and within each chapter we will make additional definitions specific to that chapter as needed. The majority of our concepts are standard and further detail can be found in most books on graph theory such as [23].

1.1.1 Basic definitions

A graph G is an object comprised of a set of vertices $V(G)$ and a set of edges $E(G) \subseteq \{u \times v \mid v \in V(G), u \in V(G)\}$. We follow the usual convention and say that the *order* of G is the cardinality of the vertex set and the *size* of G is the cardinality of the edge set. A graph is called *undirected* if the edges are unordered, that is, the edge (u, v) is the same as the edge (v, u). Otherwise the graph is *directed*. A graph is *simple* if there is at most one edge between any pair of vertices, and there is no edge connecting a vertex to itself. Unless otherwise stated, every graph considered in this thesis is assumed to be simple and undirected.

For an undirected graph G, if $(u, v) \in E(G)$ then we say that u and v are *adjacent*. For an edge $(u, v) = e \in E(G)$, we say that vertices u and v are *incident* to e and also that e is incident to u and v. The *degree* of a vertex v, denoted as $d(v)$, is equal to the number of adjacent vertices to v and the largest degree of any vertex in G is denoted as $\Delta(G)$. The set of all adjacent vertices to $v \in V(G)$ is called the *neighbourhood* of v and is denoted as $N_G(v)$. If the graph to which the neighbourhood belongs is clear from context, then we shall drop the subscript G. Thus $d(v) = |N(v)|$.

A graph G is *k-regular* if every vertex has degree k. In the special case that a graph is 3-regular, it is called a *cubic graph*.

A *walk* in a graph G is a list of vertices $v_1, v_2, \ldots, v_k$ such that $(v_i, v_{i+1}) \in E(G)$ for each $i = 1, 2, \ldots, k-1$. A *path* is a walk such that each of the vertices are unique. A walk is *closed* if it begins and terminates at the same vertex, that is, $v_1 = v_k$. In the case that a walk is closed and each vertex other than the first (last) vertex is unique, then it is called a *cycle*. The length of a walk (path) $v_1, v_2, \ldots, v_k$ is $k-1$. Note that in the case of a cycle, the length is the number of vertices in the cycle.

A simple graph G is *connected* if for any two vertices $u, v \in V(G)$ there exists a walk between u and v. Otherwise the graph is *disconnected*. A *connected component* (sometimes simply called a component) of a simple graph is a set $S \subseteq V(G)$ such that for any two vertices $u, v \in S$ there exists a walk between u and v and for any vertices $v \in S$ and $w \in V(G) \setminus S$, there does not exist a walk between v and w. Thus a connected graph has just the one connected component and a disconnected graph has more than one connected component. Note that with these definitions, a single *isolated* vertex which possibly contains an edge to itself (a loop) and no other edges constitutes a connected component.

A connected graph G is *k-connected* if there does not exist a set of $k-1$ vertices in G whose deletion (along with their incident edges) disconnects the graph. Similarly a connected graph G is *k-edge-connected* if there does not exist a set of $k-1$ edges such that the deletion of those edges disconnects the graph.

A *subgraph* of a graph G is formed by a set of vertices $S \subseteq V(G)$ and a set of edges $T \subseteq \{u \times v \mid v \in S, u \in S, (u, v) \in E(G)\}$. If H is a subgraph of G, then G is also a *supergraph* of H. A *vertex induced subgraph* of G is the graph formed by a set of vertices $S \subseteq V(G)$ and a set of edges $T = \{(u, v) \mid v \in S, u \in S, (u, v) \in E(G)\}$. An *edge induced subgraph* of G is the graph formed by a set of edges $T \subseteq E(G)$ and the set of vertices $S = \{u \mid \exists v \in T, (u, v) \in T\}$.

Two graphs G and H are *isomorphic* if there is a bijective function f between the vertex sets of G and H such that any two vertices $u, v \in V(G)$ are adjacent if and only if $f(u)$ and $f(v)$ are adjacent in H. We shall use $G \cong H$ to denote that G and H are isomorphic.

An *edge contraction* is an operation on an edge (u, v) where the edge (u, v) is deleted and the vertices u and v are identified into a new vertex. A graph H is a *minor* of a graph G if, there is some sequence of edge deletions and edge contractions in G which results in a graph that is isomorphic to H.

A graph is *planar* if it can be drawn into the plane in such a way that its edges only intersect at their endpoints. Note that the plane is homeomorphic to the surface of a sphere, minus one point. Hence if a graph is planar, then the famous polyhedron formula of Euler holds. That is,

$$|V(G)| - |E(G)| + F = 2, \tag{1.1}$$

where F is the number of faces of the corresponding polyhedron. There is a limit to the number of edges which a planar simple graph can have. This can be seen by noting that each edge is on the boundary of at most two faces and each face is bounded by at least three edges. Hence, if we count each edge twice, we are counting each face at least three times and so, $2|E(G)| \geq 3F$. Then multiplying (1.1) by 3,

$$3|V(G)| - 3|E(G)| + 3F = 6,$$

and substituting in $2|E(G)| \geq 3F$, we obtain an upper bound on the number edges in any planar graph,

$$3|V(G)| - 3|E(G)| + 2|E(G)| \geq 6,$$

$$|E(G)| \leq 3|V(G)| - 6.$$

One of the seminal results in relation to planar graphs is Kuratowski's Theorem, which provides a complete characterisation of planar graphs:

Theorem 1.1 (Kuratowski [99]). *A graph G is planar if and only if neither K_5 nor $K_{3,3}$ are minors of G.*

1.1.2 Graph products

Some graphs can be thought of as resulting from specific graph products, and we outline two such products here that will be used in this thesis.

The *Cartesian product* of two graphs G_1 and G_2, denoted as $G_1 \square G_2$, is the graph whose vertex set is $\{(v_1, v_2) \mid v_1 \in V(G_1), v_2 \in V(G_2)\}$ and two vertices (u_1, u_2) and (v_1, v_2) are connected with an edge if and only if one of the following two conditions is satisfied:

1. $(u_1, v_1) \in E(G_1)$ and $u_2 = v_2$, or
2. $u_1 = v_1$ and $(u_2, v_2) \in E(G_2)$.

The *join product* of two graphs G_1 and G_2, denoted as $G_1 + G_2$, is the graph whose vertex set is $V(G_1) \cup V(G_2)$ and edge set is $E(G_1) \cup E(G_2) \cup \{(v_1, v_2) \mid v_1 \in E(G_1), v_2 \in E(G_2)\}$.

1.1.3 Asymptotic notation and complexity

Let $g(x)$ be a function defined on some unbounded set of positive reals, and suppose that there exists a constant M such that $g(x) > 0$ for all $x \geq M$. We say that $g(x) = O(f(x))$ if there exists constants $C > 0$ and N such that $|g(x)| \leq Cf(x)$ for all $x \geq N$. Intuitively, this implies that $|g(x)|$ is bounded above by some constant factor of $f(x)$, for all large enough x. Similarly, we say that $g(x) = \Omega(f(x))$ if there exists constants $C > 0$ and N such that $g(x) \geq Cf(x)$ for all $x \geq N$. Intuitively, this implies that $g(x)$ is bounded below by some constant factor of $f(x)$, for all large enough x. We say that $g(x) = \Theta(f(x))$ if there exists constants $C_1 > 0$, $C_2 > 0$ and N such that $C_1 f(x) \leq g(x) \leq C_2 f(x)$ for all $x \geq N$. Intuitively, this implies that $g(x)$ is bounded both above and below by some constant factor of $f(x)$, for all large enough x.

For the below discussions, given a graph G, let $n = |V(G)|$ and $m =$

$|E(G)|$. A simple graph G is a *dense graph* if $m = \Theta(n^2)$, otherwise it is a *sparse graph.* Note that the maximal number of edges in a simple graph is attained by the complete graph which has $(n^2 - n)/2$ edges. A simple application of the pigeonhole principle gives a one-way relation between a dense graph G and its maximum degree $\Delta(G)$.

Lemma 1.2. *If a graph G is dense, then $\Delta(G) = \Omega(n)$.*

Proof. By definition, there exists constants $C_1 > 0$ and $C_2 > 0$ such that

$$C_1 n^2 \leq m \leq C_2 n^2.$$

Therefore, there are at least $2C_1 n^2$ end-vertices of edges which are spread across n vertices. Then there is at least one vertex v which is incident to at least $(2C_1 n^2)/n$ edges. Hence,

$$\Delta(G) \geq d(v) \geq 2C_1 n.$$

Therefore, $\Delta(G) = \Omega(n)$. □

We shall be describing various algorithms throughout this thesis, and will be discussing the runtimes of these algorithms. An algorithm takes an input of size x and performs some number of operations before it concludes. If the algorithm concludes in $g(x) = O(f(x))$ operations, then we shall say that the algorithm takes $O(f(x))$ time. For the following discussion, an algorithm is *polynomial time* if it takes $O(f(x))$ time where f is some polynomial in x. Similarly an algorithm is *linear time* if f is linear in x and *exponential time* if f is exponential in x.

We briefly introduce some *complexity classes* related to the problems tackled in this thesis, but skip much of the detail and subtleties of this broad subject. The interested reader is referred to [60] for comprehensive discussions about complexity theory. A problem is a *decision problem* if it can be posed as a TRUE or FALSE question. A decision problem lies in the set called *non-deterministic polynomial time* (NP) if a TRUE output can be verified in polynomial time. A decision problem $\mathcal{P}$ is in NP-complete if it is in NP, and in addition, every other problem in NP can be reduced to $\mathcal{P}$ in polynomial time. Intuitively, NP-complete is the set of decision problems which can have any other NP problem reduced to it quickly, and any solution can be verified quickly, however, it is not necessarily quick to find that solution in the first place. A decision problem $\mathcal{P}$ is NP-hard if every problem in NP can be reduced to $\mathcal{P}$ in polynomial time, but a TRUE output no longer needs to be verifiable in polynomial time. Thus NP-hard decision problems are a superset of the NP-complete decision problems.

It is not known if there exists any decision problems in NP-complete which can be solved in polynomial time. This important question is connected to the famous P vs. NP problem which is unquestionably the largest open problem in this area. From the property that any NP problem can be reduced to any NP-complete problem in polynomial time, if one problem in NP-complete could be solved in polynomial time, then using such a reduction, every problem in NP could be solved in polynomial time.

As the size of the input to a problem increases, many problems in NP-complete and NP-hard become intractable for computers to solve. Many real-world scenarios can be posed as NP-complete and NP-hard problems and this is one reason why the study of heuristic methods for these problems is of current importance.

1.1.4 Embeddings, drawings and crossing numbers

The following definitions are standard in literature, and are repeated from [40].

An *embedding* of a graph G onto a surface Σ (a compact, connected 2-manifold) is a representation of G onto Σ such that vertices are distinct points on Σ and each edge e is a simple arc on Σ connecting the points associated with the end vertices of e. The embedding must also satisfy: An arc of edge e does not include any points associated with vertices other than the end vertices of e, and two arcs never intersect at a point which is interior to either of the arcs. Two embeddings are equivalent if there is a homeomorphism of Σ which transforms one into the other. The equivalence class of all such embeddings is a *topological embedding* of G.

An embedding Π is a *two cell embedding* if each of the connected components of $\Sigma - \Pi$ are homeomorhpic to an open disk. In this present work, we are only concerned with embeddings in which Σ is the surface of a sphere and so each embedding is a two cell embedding.

A topological embedding of G onto the sphere uniquely defines a cyclic ordering of the edges incident to each vertex of G and the collection of these cyclic orderings is a *combinatorial embedding* for G. A combinatorial embedding Π defines a set of cycles in G which bound the faces of any embedding belonging to the associated topological embedding, and so we may talk about the set of 'faces' of Π. Similarly, Π defines a dual graph Π^* which is isomorphic to the dual graph of any embedding belonging to the associated topological embedding. Note that the edges $e_1, e_2, \ldots, e_m$ of G are in one-to-one correspondence with edges $e_1^*, e_2^*, \ldots, e_m^*$ of the dual graph Π^*.

A *drawing* D is a representation of a graph G onto the plane with similar conditions to an embedding. Vertices are represented as distinct points, and each edge e is represented by a simple arc between the points associated with the end vertices of e. The drawing must also satisfy: An arc of edge e does not include any points associated with vertices other than the end vertices of e, and any intersection between the interiors of arcs involves at most two arcs. Given a drawing D of G, the intersections which occur in the interiors of arcs are the *crossings* of the drawing and the number of crossings is denoted by $cr_D(G)$.

Definition 1.3 (The crossing number)**.** The *crossing number* of a graph is denoted by $cr(G)$ and is the minimum number of crossings over all possible drawings of G.

In what follows we shall also describe some variants of the crossing number. To help distinguish these cases, the crossing number is sometimes referred to as the plane (or planar) crossing number. It is now clear that if $cr_D(G) = 0$, then G is a planar graph, and we say that D is a *planar drawing* of G. Then, for any non-planar drawing D, the *planarisation* of D is a planar drawing of the planar graph obtained by replacing crossings of the initial drawing with dummy vertices of degree 4. Hence the graph corresponding to the planarisation of D has $n + cr_D(G)$ vertices and $m + 2cr_D(G)$ edges. Figure 1.2 provides a simple example of a drawing D of a graph, which has 1 crossing, and a planarisation D' of D. The combinatorial embedding corresponding to D' can be represented by the collection of cyclic orderings (clockwise) of edges around each vertex:

$v_1 : \{e_1, e_6, e_4\}$
$v_2 : \{e_5, e_1, e_3\}$
$v_3 : \{e_4, e_7, e_2\}$
$v_4 : \{e_2, e_8, e_3\}$
$v_5 : \{e_5, e_8, e_7, e_6\}$

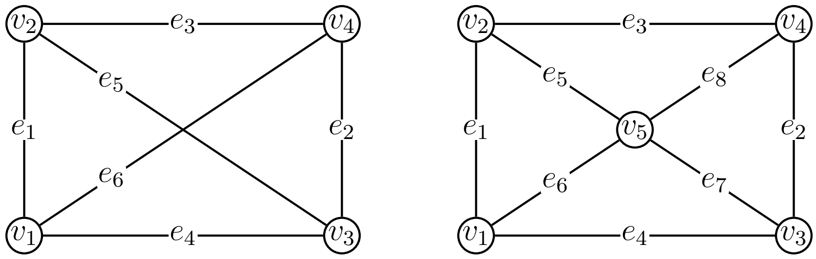


Figure 1.2: On the left, a drawing of a graph and on the right, a planarisation of the drawing.

In what follows, when no confusion is possible, we shall refer to the arcs of a drawing or embedding as 'edges' of the drawing (or embedding) and the points associated with vertices as 'vertices' of the drawing (or embedding).

Definition 1.4. A drawing is a *good drawing* if the edges of the drawing satisfy the following conditions:

1. No two edges cross each other more than once.
2. No two edges which share a common end vertex cross.
3. No edge crosses itself (note that this trivially holds from the definition of a simple arc).

The following is a simple exercise, which is discussed without proof in [89] and shows that any drawing which attains $cr(G)$ crossings is a good drawing of G. We include a proof here for the sake of completeness.

Lemma 1.5. *For any graph G, any drawing which attains $cr(G)$ crossings, is a good drawing.*

Proof. Consider a drawing D of G which is not a good drawing and suppose that D has $cr(G)$ crossings. We work through the conditions of Definition 1.4 and show that for each condition, there exists a modification to D with fewer crossings, violating the assumption that D has $cr(D)$ crossings.

(1) Suppose D has a pair of edges, e_1 and e_2, which cross more than once. We may reroute e_1 and e_2 as in Figure 1.3 so that the e_1 now crosses exactly those edges that e_2 crossed within the rerouted sections and vice versa. Rerouting the edges in this removes at least one crossing from D, and the number of crossings in the rest of D remains unchanged.

(2) Suppose now that D has a crossing between two edges, e_1 and e_2, which share a common end-vertex. We may reroute e_1 and e_2 as in Figure 1.4 so that e_1 now crosses exactly those edges that e_2 crossed within the rerouted sections and vice versa. Clearly the number of crossings between e_1 and e_2 have reduced and the number of crossings in the rest of D remains unchanged.

(3) Finally, suppose an edge e_1 of D crosses itself, then we may reroute e_1 as in Figure 1.5 so that the number of crossings is reduced. □

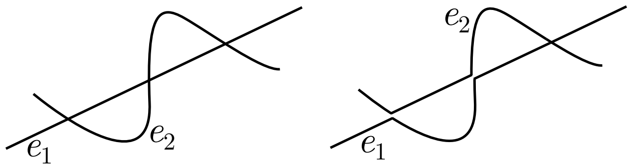


Figure 1.3: Modifying edges e_1 and e_2 we remove at least one crossing between e_1 and e_2 and the number of crossings in the rest of D remains unchanged.

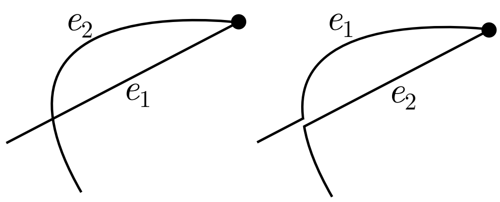


Figure 1.4: Modifying edges e_1 and e_2 removes one crossing between e_1 and e_2 and the number of crossings in the rest of D remains unchanged.

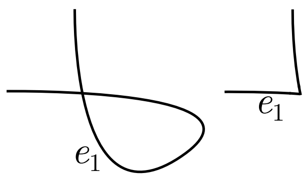


Figure 1.5: Modifying e_1 removes at least one crossing from D.

1.2 Crossing numbers

Crossing numbers and crossing minimisation has an enormous body of literature and there are several valuable resources to help navigate this. The fascinating beginnings and early research on crossing numbers are recounted in [20]. For over a decade, Imrich Vrt'o has maintained an extensive bibliography of research relating to crossing numbers [139]. In 2018, Markus Schaefer released a (perhaps overdue) book detailing some of the successful approaches to studying crossing numbers [125]. Schaefer also maintains a detailed survey on the many different variants of the crossing number at [126].

A discussion on some of the successful research focusing on algorithmically computing crossing numbers is given in [26]. Finally, a recently released comprehensive survey on all graphs and graph families with known crossing numbers is given in [39]. In the following, we only scrape the surface of some results which have provided decades of fruitful research.

Lower bounds on the crossing number of general graphs were originally conjectured by Erdős and Guy in 1973 [53]. By placing a condition on the number of edges, this was subsequently proved by Ajtai et al. in 1982 [9] who provided a lower bound on the number of crossings which depends upon a constant c.

Theorem 1.6 (Ajtai et al., 1982 [9])**.** *There is a constant $c > 0$ such that for any graph G on n vertices and $m \geq 4n$ edges,*

$$cr(G) > \frac{cm^3}{n^2}.$$

Ajtai et al. originally proved Theorem 1.6 for $c = \frac{1}{100}$. Later, this was improved to $c = \frac{1}{64}$ by Chazelle et al. in an email conversation summarised in [8]. Different values of c as well as different conditions on the number of edges have been considered closely in a line of research which is summarised in Ackerman [5]. The most recent result, by Ackerman in 2015 (revised 2019) [5], showed that if $m \geq 6.95n$, then c can be increased to $\frac{1}{29}$.

The conjectured values for the crossing numbers of the complete graphs and the complete bipartite graphs are interesting in their own right, but they have also been utilised to derive many other theoretical results. For this reason, the conjectured values have been assigned specific notation as functions. We define

$$H(n) := \frac{1}{4}\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor\left\lfloor\frac{n-2}{2}\right\rfloor\left\lfloor\frac{n-3}{2}\right\rfloor,$$

and similarly,

$$Z(n_1, n_2) := \left\lfloor \frac{n_1}{2} \right\rfloor \left\lfloor \frac{n_1 - 1}{2} \right\rfloor \left\lfloor \frac{n_2}{2} \right\rfloor \left\lfloor \frac{n_2 - 1}{2} \right\rfloor.$$

Then, the original conjectures, named after Hill and Zarankiewicz are as follows.

Conjecture 1.7 (Hill's Conjecture)**.** *Consider the complete graph on n vertices K_n, then*

$$cr(K_n) = H(n).$$

Conjecture 1.8 (Zarankiewicz' Conjecture)**.** *Consider the complete bipartite graph on $n_1 + n_2$ vertices K_{n_1,n_2}, then*

$$cr(K_{n_1,n_2}) = Z(n_1, n_2).$$

From a mixture of theoretical arguments and brute force computations, Conjecture 1.7 is now known to hold for values of $n \leq 12$ [69, 113]. The next unknown case $cr(K_{13})$ has been considered in [109, 3] and narrowed down to one of two remaining possible values, either 223 or 225. Conjecture 1.8 is known to hold for $\min\{n_1, n_2\} \leq 6$ [89, 71], as well as the specific cases where $7 \leq n_1 \leq 8$ and $n_1 \leq n_2 \leq 10$ [143].

The problem of determining the crossing number in general graphs is known to be an NP-complete problem [61]. The problem remains NP-complete even when restricted to certain families of graphs. Known restrictions include the family of cubic graphs [79] and the family of planar graphs with the addition of a single edge [28]. Interestingly, the crossing number is an example of a fixed parameter tractable problem [64]. The fixed parameter problem can even be solved in linear time [88], however, the known algorithms to solve the fixed parameter problem are of no practical use.

1.2.1 Variants of the crossing number

There are many interesting variants of the crossing number which have been studied and these are comprehensively detailed in [126]. These notions have arisen from both theoretical and practical perspectives and we now describe a few of the widely studied variants.

Book crossing number

A *book* of k 'pages' is a collection of k half-planes with all of their boundary lines identified to form the 'spine' of the book. Graph drawings on a book are restricted to the following: All vertices lie on the spine of the book, and each edge is contained entirely in a single page.

The book crossing number on k pages, $bcr_k(G)$ is the minimum number of crossings of any drawing of G which obeys the above restrictions. It is known that $bcr_2(K_n) = H(n)$ [2] and in general De Klerk et al [44] provides a two parameter function $f(k, n)$ such that $bcr_k(K_n) \leq f(k, n)$ for all integers $k, n > 0$ and they conjecture that equality holds.

Genus crossing number

The genus crossing number extends the notion of the usual crossing number by considering drawings onto surfaces of different genus. Let Σ_g be a surface of genus g. Given a graph G, the minimum number of crossings of any drawing of G onto Σ_g is denoted as $cr_{\Sigma_g}(G)$. It is known that $cr_{\Sigma_g}(G) = \Omega(m^3/n^2)$ if $0 \leq g < n^2/m$ and $cr_{\Sigma_g}(G) = \Omega(m^2/n)$ if $n^2/m \leq g \leq n/64$ [128].

Maximum crossing number

For a graph G, consider the drawings of G such that edges with a common end-vertex do not cross and each pair of edges crosses at most once. Then the maximum number of crossings in any such drawing is the *maximum crossing number* of G, denoted $maxcr(G)$. It is known that $maxcr(K_n) = \binom{n}{4}$ [122] and that $maxcr(K_{n_1,n_2,\ldots n_k}) = \binom{p}{4} - \sum_{i=1}^{k}(\binom{n_i}{4} + (p - n_i)\binom{n_i}{3})$, where $p = \sum_{i=1}^{k} n_i$ and $k \geq 2$ [76]. In [12], Archdeacon conjectures that $maxcr(H) \leq maxcr(G)$ for any subgraph $H \subseteq G$. This problem remains open and is in contrast to the analogous problem for the standard crossing number, in which the result is obtained by a simple observation. The maximum crossing number is also closely related to Conway's thrackle conjecture [104] which postulates that if G admits a drawing such that every pair of edges either: has a common end-vertex, or crosses exactly once, then $|E(G)| \leq |V(G)|$.

Rectilinear crossing number

The rectilinear crossing number, denoted $\overline{cr}(G)$ is the smallest number of crossings in any drawing of G onto the plane such that edges are all straight line segments. Clearly $\overline{cr}(G)$ constitutes an upper bound for $cr(G)$, that is, $cr(G) \leq \overline{cr}(G)$. It is also known that in some cases the inequality is strict, for example, $18 = cr(K_8) < \overline{cr}(K_8) = 19$ [17, 131]. Indeed, for the complete graphs, it is known that the inequality is strict for $n = 8$ and all $n \geq 10$ [1]. The exact value of $\overline{cr}(K_n)$ has been computed for all $n \leq 27$ [7] and efforts continue towards extending this to larger values.

1.3 Motivating example

In many cases, the exact crossing number of a graph is found by obtaining agreeable upper and lower bounds. The upper bound is usually found by way of a drawing construction and the lower bound is usually obtained theoretically. We now demonstrate one example of finding agreeable upper and lower bounds in order to obtain the crossing number of the famous Petersen graph.

Recall that the girth of a graph is the length of its shortest cycle. The following lower bound was utilised in the 1960's and was most probably noticed earlier.

Lemma 1.9. *Let G be a finite graph without loops or parallel edges. Let $n = |V(G)|$, $m = |E(G)|$ and g be the girth of G. Then*

$$cr(G) \geq \left\lceil m - \frac{g}{g-2}(n-2) \right\rceil . \tag{1.2}$$

Proof. Let D be an optimal drawing of G with $cr(G)$ crossings and consider iteratively removing, both from the drawing and from the graph, those edges which are crossed. If we remove all such edges, then we will arrive at a planar drawing of some planar graph. Let h be the smallest possible number of edges whose removal produces a planar graph and let G' denote the resulting planar graph. Let $n = |V(G')| = |V(G)|$, and $m' = |E(G')|$, thus $m' = m - h$. Euler's polyhedron formula holds for G',

$$n - m' + F = 2, \tag{1.3}$$

where F is the number of faces in an embedding of G'. Let g and g' be the girths of G and G' respectively and note that by deleting edges from G, we cannot decrease the girth, thus $g' \geq g$. Each face of an embedding of G' is bounded by a cycle and hence the following holds

$$Fg' \leq 2m'. \tag{1.4}$$

Multiplying (1.3) by g', we get

$$ng' - m'g' + Fg' = 2g',$$

and then utilising (1.4),

$$ng' - m'g' + 2m' \geq 2g',$$

$$g'(n-2) \geq m'(g'-2),$$

$$\frac{g'(n-2)}{g'-2} \geq m' = m - h.$$

Using the fact that $g' \geq g$, we arrive at

$$h \geq m - \frac{g'(n-2)}{g'-2} \geq m - \frac{g(n-2)}{g-2}.$$

There are at least h crossings in D, possibly more, and so,

$$cr(G) \geq h \geq m - \frac{g(n-2)}{g-2}.$$

Finally, the crossing number is an integer and so the ceiling function can be applied. □

If the girth and the edge density of the graph are sufficiently large, then the lower bound given by Lemma 1.2 can be applied to find the crossing number. For example, Figure 1.6 displays two drawings of the Petersen graph P, the first with five crossings and the second with two crossings. It can be checked that the girth of the Petersen graph is 5 and so the lower bound from Lemma 1.2 provides $cr(P) \geq \left\lceil \frac{5}{3} \right\rceil = 2$. The drawing on the right in Figure 1.6 contains two crossings and so $cr(P) \leq 2$. Therefore we obtain equality.

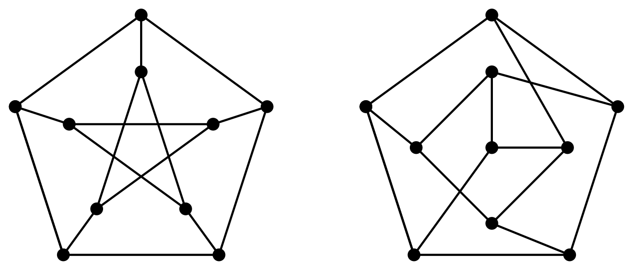


Figure 1.6: The Petersen graph drawn first with 5 crossings, and second with 2 crossings, which is a crossing optimal drawing.

In some cases obtaining drawings of graphs which possess a given number of crossings can be deceptively difficult. Indeed, finding general drawings of the n-dimensional cube which possess a small number of crossings have been the subject of ongoing research for more than 30 years. Recent breakthroughs were made in [55], however, further improvements in [145] imply that better upper bounds may still be obtainable.

Lastly, we remark that Lemma 1.9 was implicitly extended for use in graphs drawn on surfaces of genus one by Guy in [72], and in [86], Kainen generalised the result to graphs drawn on surfaces of arbitrary genus.

Theorem 1.10 (Kainen, 1972 [86])**.** *Let G be a finite simple graph. Let $n = |V(G)|$, $m = |E(G)|$ and g be the girth of G. Then for any surface Σ_δ of genus δ,*

$$cr_{\Sigma_\delta}(G) \geq m - \frac{g}{g-2}(n - 2(1-\delta)),$$

where $cr_{\Sigma_\delta}(G)$ is the genus crossing number of G on Σ_δ.

Chapter 2

A new crossing minimisation heuristic based on star insertion

In this chapter we present a new heuristic method for minimising crossings in a graph. The method is based upon repeatedly solving the so-called *star insertion problem* in the setting where the combinatorial embedding is fixed, and has several desirable characteristics for practical use. We begin by reviewing the most successful exact and heuristic methods for crossing minimisation with a focus on those algorithms which have been implemented. Next, we introduce our new proposed method and discuss some aspects of algorithm design for our implementation. We then perform comparative experiments using our implementation and provide some experimental results. The results indicate that our method compares well to existing methods, and also that it is suitable for dense instances.

2.1 Related work

We first review some exact and heuristic methods for crossing minimisation in general graphs. We focus only on algorithms that have been designed for the plane crossing number. There has also been significant effort in developing algorithms for the many variants of the crossing number. However, reviewing these is beyond the scope of this thesis and instead we just make the observation that some, but not all, of these methods can also be used to provide an upper bound for the plane crossing number. For example, if a heuristic provides a rectilinear drawing of a graph G which possesses c crossings, then c is also an upper bound for the plane crossing number of G. Conversely, if a heuristic provides a 3-page book drawing of a graph G which possesses c crossings, then the drawing does not necessarily provide an upper bound of c for the plane crossing number of G. For a review of the some of the important algorithms for the variants of the crossing number, we direct the reader to the masters thesis of Winterbach [142].

2.1.1 Exact algorithms

To date, there have been three main methods for computing exact crossing numbers of general graphs. For each of these, we give a brief description and discuss an implementation which has been reported on.

Garey and Johnson algorithm

In [61], Garey and Johnson describe a brute force approach to computing the crossing number of an arbitrary graph. Consider a graph G with $|V(G)| = n$ and $|E(G)| = m$ and a fixed integer k. An interpretation of the Garey and Johnson algorithm proceeds as follows. First, each edge of G is sub-divided $k = 1$ times and the resulting graph is called G'. Thus each edge of G becomes a 'chain' of k edges in G'. Next, choose k sets of pairs of edges from $E(G')$. There are combinatorially many ways to choose these pairs. Each chosen pair will hopefully correspond to a crossing in some drawing of G'. So, each chosen pair of edges is replaced by a dummy vertex along with four new edges connecting to their endpoints. The resulting graph is then checked for planarity. If every possible combination of k pairs of edges is exhausted, and no such planar graph has been obtained, then k is incremented and the above process is repeated. If the above is performed for $k = 0, 1, 2, 3, ...$, then the first time that a planar graph is obtained gives the value for $cr(G)$. The planarity check can be done performed efficiently, indeed in linear time with respect to the number of vertices in the modified graph.

Notice that there are $p = \binom{k|E(G)|}{2}$ distinct pairs of edges in G'. Thus, for a given k, the number of times that the planarity check must be performed can be bounded above by $\binom{p}{k}$. The number of vertices in the modified G' is $n+mk$, and so the runtime of the above process is $O(\binom{p}{k}(n + mk))$. This is clearly a fairly naive design and there are some obvious shortcuts that may be taken advantage of. For example, there are many unnecessary vertices remaining in the modified G'. It is also possible to impose that the drawing corresponding to the selected k-tuple, is a good drawing, thus reducing the number of k-tuples which need to be considered. Several methods for improving the above algorithm have been discussed and implemented in [142], however, it is noted that the overall structure and runtime of the algorithm remains very similar to that which is described above. An implementation of the algorithm, along with some improvements, were given in [142]. The implementation was able to find $cr(K_6)$, however, $cr(K_7)$ was not tractable. We note that the algorithm given in [142] was not the focus of the manuscript and we suspect that with some specially designed data structures, K_7 and beyond would become tractable.

Harris and Harris algorithm

In [77], Harris and Harris propose a branch and bound method to construct an optimal drawing of the graph in an exhaustive manner. Their method utilises *rotational embedding schemes* to represent an embedding of a graph onto a surface. For a graph G, a rotational embedding scheme is a set of permutations $(\pi_1, \pi_2, ..., \pi_n)$ such that, for a vertex i, π_i consists of the vertices in the neighbourhood of i. Rotational embedding schemes are analagous with our definition of combinatorial embeddings, which play an important role in our heuristic in Section 2.2. A well-known theorem, see [29], relates rotation schemes to an embedding of a graph onto a surface.

Theorem 2.1. *For each 2-cell embedding of G onto some surface, there exists a unique tuple $(\pi_1, \pi_2, ..., \pi_n)$, where each π_v is a permutation which describes the cyclic ordering, in the embedding, of vertices adjacent to vertex v. Conversely, for each tuple of permutations $(\pi_1, \pi_2, ..., \pi_n)$, there exists a 2-cell embedding of G onto some surface, such that π_v describes the cyclic ordering, in the embedding, of vertices adjacent to vertex v.*

The Harris and Harris algorithm begins with an empty graph, and draws edges one at a time, whilst preserving planarity until no more edges can be drawn without destroying the planarity. At this stage the algorithm has constructed a locally maximal planar subgraph. From here, each remaining edge is drawn in, one at a time, producing crossings. Once all edges have been introduced, the algorithm backtracks and attempts to draw edges with fewer crossings until it has exhaustively determined a drawing with the fewest possible crossings. The exhaustive search is achieved with a branch and bound model. Once an edge has been introduced in the branch and bound stage, the newly introduced crossings are planarised in order to preserve the planarity of the underlying graph, on which the algorithm is modifying. Thus, at each step of the algorithm an embedding of the partial (planarised) graph is maintained, and this allows for the modifications to be performed by primarily utilising a corresponding rotational embedding scheme. A parallelised implementation of the algorithm by Tadjiev and Harris in [136] was able to find $cr(K_8)$, and from the runtime, it is clear that complete graphs beyond K_8 would require additional specialised methods.

During the Harris and Harris algorithm, every possible good drawing of the graph is considered. If complete graphs are the sole aim of the algorithm, then restrictions on the type of drawings produced during the intermediate steps can be introduced. Fredrickson et al. [59] investigated these restrictions and were able to reduce the computation time significantly for the complete graphs. With their improved method, it is noted in [59], that $cr(K_8)$ can be found after a few hours, however, $cr(K_9)$ remained beyond their reach.

Integer linear programming formulation

In recent years, sophisticated integer linear programming (ILP) approaches for finding $cr(G)$ have been developed in [26], [30] and [33]. These methods have been reported to be the most successful in solving both dense and sparse graphs. We provide a brief overview and direct the reader to [26] for a detailed survey of the current results. The ILP is constructed so that each variable corresponds to a potential crossing in some drawing of G. The constraints of the ILP ensure that the feasible solutions define a planarisation of some drawing of G. Hence, a solution with the minimal number of positive variables defines a crossing-minimal drawing of the graph. The constraints are designed to prevent any subdivision of a $K_{3,3}$ or K_5 occurring in the planarisation, and hence by Kuratowski's Theorem (Theorem 1.1), define a planar graph. One difficulty arising from such a formulation is that these constraints are combinatorial in number. Another difficulty is that there is no simple way to handle the situation in which an edge is required to be crossed more than once. The latter difficulty was originally resolved by subdividing the edges of G many times. Each edge of G was subdivided into k edges, where k is an upper bound on the maximum number of times an edge may be crossed in an optimal solution. A very simplistic bound for k is $|E(G)|-d(v)-d(u)-1$, because in any optimal solution, an edge never crosses itself, or any edges incident to one of its end-vertices. These subdivisions ensure that edges only need to be crossed at most once in the solution.

However, doing this also requires a large number of variables ($\Omega(|E(G)|^4)$) and so an alternative method is recommended in [33] and reported to be the current best performing formulation. Additional variables are used, from which the order of the crossings along any edge can be deduced. This is done specifically to resolve the problem of an edge being crossed more than once. To handle the combinatorial number of constraints, a branch-and-cut-and-prize method is used on the linear relaxation of the ILP. The extra variables can then take advantage of the branch-and-cut-and-prize framework, by only being introduced to the model when they are required. Experiments and comparisons can be found in [33], and in [111] Mutzel explains that their implementation is able to solve K_{12}, however, K_{13} remains too difficult for their current methods.

2.1.2 Heuristic methods

Crossing minimisation has been considered in a number of contexts. Exact methods are often not able to solve practical problems and instead heuristic methods are utilised. For example, in the field of automated graph drawing, heuristics have been developed to construct drawings of graphs or networks with desirable characteristics, one of which is a low number of crossings. Approaches including force-directed drawing algorithms [50, 73, 87] and genetic algorithms [24, 52, 16] have been developed for this purpose. When crossing minimisation is the sole aim, arguably the most successful heuristics to date have been based on edge insertion procedures.

Edge insertion and the Planarisation method

In general, insertion procedures refer to the act of constructing a drawing of a graph by starting with some partial drawing and then adding the missing elements to it. Sophisticated algorithms based on insertion procedures have been developed and we discuss some of these below.

Throughout this chapter, we will often consider the situation where we have a combinatorial embedding Π of a graph, and then need to add an edge e to the graph and obtain an updated combinatorial embedding. In such cases, we will say that we are *inserting* an edge e into Π, as follows. Suppose that $e = (v_1, v_2)$, where $v_1, v_2 \in V(G)$, and let Γ be an embedding which realises the cyclic orderings in Π. A simple arc connecting v_1 and v_2 may be added to Γ, such that the interior of the arc intersects only with the interiors of some (possibly empty) ordered set of edges $\{e_1, e_2, \ldots, e_k\}$ already present in Γ. Clearly, for any embedding which realises the cyclic orderings in Π, such an arc can be found which intersects exactly the same set of edges $\{e_1, e_2, \ldots, e_k\}$. We also refer to these intersections as 'crossings'.

The Edge Insertion Problem (EIP), which is studied in [67], has two variations depending on the definition of optimality used; the fixed embedding variation and the variable embedding variation:

Definition 2.2. (EIP – fixed embedding) Given a combinatorial embedding Π of a graph G and a pair of vertices $v_1, v_2 \in V(G)$, insert the edge $e = (v_1, v_2)$ into Π in such a way that the number of crossings is minimised.

Definition 2.3. (EIP – variable embedding) Given a planar graph G and a pair of vertices $v_1, v_2 \in V(G)$, find a combinatorial embedding Π of G such that inserting the edge $e = (v_1, v_2)$ into Π so as to minimise the crossings results in the minimal number of crossings among all embeddings of G.

The fixed embedding problem can be solved in $O(n)$ time by finding a shortest path in a modified dual graph, and this is explained in detail in [65]. In [67] it is shown that the variable embedding problem can also be solved in $O(n)$ time by taking advantage of the properties of maximal triconnected components and SPQR trees. Note that solving the edge insertion problem is subtly different from computing the crossing number of $G + e$

(the graph G with the addition of edge e). However, it is shown in [80] that the number of crossings introduced in a solution to the variable embedding version approximates $cr(G + e)$ to within some factor and the best possible factor is proved in [27] to be $\lfloor \Delta(G)/2 \rfloor$.

The planarisation method, a highly effective crossing minimisation heuristic, is based upon repeatedly solving the edge insertion problem. In particular, the planarisation method involves attempting to solve two separate problems:

1. Compute a planar subgraph G_p of G - ideally a maximum planar subgraph.
2. Iteratively re-insert the remaining edges of G into a combinatorial embedding of G_p while striving to keep number of crossings as small as possible.

Computing a maximum planar subgraph is NP-hard [103], so instead a locally maximal planar subgraph is usually used for step 1, which can be computed in $O(n + m)$ time [47].

To achieve step 2, given a planar subgraph of G, EIP (in either the fixed or variable embedding) is solved for one of the missing edges. Then any introduced crossings are replaced by degree 4 dummy vertices to obtain a new planar graph, and EIP is solved again for another missing edge, and so on until a planarised drawing of the full graph is obtained.

The planarisation method was first described in the context of EIP-fixed by Batini et al. [18]. Later, in Gutwenger [65], the method was rigorously developed for EIP-variable, along with an implementation and experimental results which were also reported in Gutwenger and Mutzel [66]. In most cases, the method based on EIP-variable provided superior solutions for the tested graphs. However, it was observed that the EIP-variable method often suffered in runtime in comparison to EIP-fixed implementations, due to the many SPQR trees which need computed (a new SPQR tree for every edge inserted). Later, in Chimani and Gutwenger [31], implementations were also reported on which focused on improving the post processing schemes that can be utilised when running these methods, and again improved results were obtained from those previously reported.

A related approach to the planarisation method is the *multiple edge insertion* problem (MEI), which involves inserting several edges simultaneously into a planar graph. Let F be the set of edges being inserted into some planar graph G. For general F, solving MEI to optimality is NP-Hard [149], and approximation algorithms have been developed in [37] and [35]. An approximate solution to MEI is known to approximate the crossing number of the graph $G + F$ [36] and so for graphs of bounded degree and bounded $|F|$, the algorithm in [37] constitutes a multiplicative factor approximation algorithm for $cr(G+F)$ and the algorithm in [35] constitutes an additive factor approximation algorithm for $cr(G + F)$. Among implementations based on MEI, only the algorithm of Chimani [35] has been experimentally reported on. In particular, it was considered in Chimani and Gutwenger [31], which is the most recent analysis on the practical usage of various crossing minimisation heuristics. Chimani and Gutwenger [31] claim that the MEI implementation from [35] achieves roughly comparable solution quality to the best iterative EIP-variable method, with the benefit of significantly reduced runtimes. If runtimes are disregarded, the iterative EIP-variable method with the addition of a significant post processing step usually produced the best solutions, however overall (in terms of both solution quality and runtime) Chimani and Gutwenger [31] advocate that the MEI implementation from [35] was the best heuristic for practical use.

Methods based on star-insertion

A natural extension to the above is the *star insertion problem* (SIP) where instead of a single edge, the object to be added to G is a vertex v along with a set of incident edges of v (collectively, a star). As before, there are fixed embedding and variable embedding versions of SIP:

Definition 2.4. (SIP – fixed embedding) Given a combinatorial embedding Π of a graph G and a vertex $v \notin V(G)$ (along with a set of incident edges whose other endpoints are all in $V(G)$), insert v along with its incident edges into Π in such a way that the number of crossings is minimised.

Definition 2.5. (SIP – variable embedding) Given a planar graph G and a vertex $v \notin V(G)$ (along with a set of incident edges whose other endpoints are all in $V(G)$), find a combinatorial embedding Π of G such that inserting v along with its incident edges into Π so as to minimise the crossings results in the minimal number of crossings among all embeddings of G.

The fixed embedding version can be solved in $O(d(v)n)$ time using a method similar to the single edge insertion version [34]. We will make use of this approach during our heuristic, and we briefly outline our implementation of this method in Section 2.2. The complexity of the variable embedding version was in question for a short time but was resolved by Chimani et al. [34] who showed it to be solvable in polynomial time by method which is briefly outlined below. Again, the number of crossings introduced in a solution to the variable embedding version is shown in Chimani, Hliněný and Mutzel [36] to approximate the crossing number of the graph $G + v$, to within a factor of $d(v)\lfloor\Delta(G)/2\rfloor$.

The approach to solving SIP-variable, described by Chimani et al. [34], can be summarised as follows for a given graph G and vertex v to be inserted.

1. Compute an SPQR tree T of G, and consider a face f in one of the skeleton graphs of T (f belongs to a set of 'interesting' faces).

2. Solve a dynamic program whose solution advises the best combinatorial embedding which admits the minimal number of crossings when inserting v into f.

3. Repeat the above for all 'interesting' faces and select the solution which results in the fewest crossings.

Although the runtime of the algorithm provided in [34] is polynomial, it is considerably higher than solving EIP-variable, and experimental results have yet to be reported on. Nonetheless, a heuristic analogous to the planarisation method, but using star insertion rather than edge insertion, could be proposed. Indeed, in Chimani and Gutwenger [31], it is asked whether a heuristic based on star insertion could compare to the proven practical performance of the heuristic methods based on edge insertion. This present work seeks to answer this question, at least for SIP-fixed, but the approach we advocate is different in character to the planarisation method.

Heuristics based on genetic algorithms

There has been some effort to develop algorithms using the genetic algorithm framework which include number of edge crossings as one of the characteristics being optimised. These genetic algorithms use a population of candidate graph drawings and optimise by manipulating the candidates in an attempt to improve a set of characteristics. Once the candidates have been manipulated, those with the poorest characteristics are removed from the population and the process is repeated. Formulations which consider edge crossings as a characteristic to optimise have been developed in [16], [24] and [52]. A main strength of the genetic algorithm framework lies in its ability to simultaneously consider many characteristics for optimising. Although there are several different implementations which consider edge crossings, the results indicate that when crossing minimisation is the sole aim, the genetic algorithms do not perform as well as other specialised crossing minimisation heuristics, and so we do not consider them further here.

2.2 Proposed heuristic method

While the philosophy of the planarisation method is to start with a planar subgraph and increase the number of crossings at each iteration as the full graph is rebuilt, our approach works in the opposite direction; we start with a combinatorial embedding corresponding to a, presumably suboptimal, drawing of the full graph and at each iteration we attempt to find a combinatorial embedding corresponding to a drawing with fewer crossings. Unlike the planarisation method, the heuristic we propose does not require a planar subgraph to be computed. Instead it relies upon iteratively solving the star insertion problem in a combinatorial embedding which corresponds to the current (non-planar) drawing of G. With the intention of keeping the new heuristic highly practical, each iteration is performed on a fixed combinatorial embedding; this is discussed further in Section 2.3.

The approach that we advocate is to iteratively obtain improved drawings of a graph in the following way. For a given drawing D of a graph G, we attempt to find a vertex v in G satisfying the following: if we remove v, and then reintroduce v by solving the star insertion problem in a corresponding (fixed) combinatorial embedding, a drawing D_2 can be obtained such that $cr_{D_2}(G) < cr_D(G)$. If there are no vertices in the graph for which this is possible, we say that the drawing D is *locally crossing-optimal*. In what follows, we will prove the following.

Theorem 2.6. *Let G be a graph containing n vertices and m edges, and D be a drawing of G which contains k crossings. There exists an algorithm that finds a locally crossing-optimal drawing D^* of G in $O((k+n)km)$ time.*

It is our contention that the number of crossings in such a D^* found by our algorithm is, typically, close to the crossing number of G. We provide experimental results justifying this assertion in Section 2.5.

Let D be some drawing of G and let D' be its planarisation. Then D' can be mapped to an embedding on the sphere, and this realises a particular combinatorial embedding. In this sense, we say that the combinatorial embedding 'corresponds' to the drawing D. Note that given such a combinatorial embedding, a drawing which is equivalent to D can be retrieved by using any planar graph drawing techniques, such as [58] or [129].

Let D be a drawing of G and let Π be a combinatorial embedding corresponding to D. Consider deleting from G a vertex v and its set of incident edges; it is clear that a subdrawing $D - v$ can be easily obtained from D. Then a combinatorial embedding corresponding to the subdrawing $D - v$ can be computed by iteratively merging faces of Π which share an edge associated with one of the deleted edges. We shall call this the *reduced combinatorial embedding* corresponding to subdrawing $D - v$ and denote it as $\Pi - v$.

We define a star insertion into a combinatorial embedding Π by utilising definitions similar to those in [67]. Let Π be a combinatorial embedding of G, let f be a face of Π and let v be a vertex of G. Then $e_1, e_2, \ldots, e_j$ is an *insertion path* for v and f if either $j = 0$ and v is on the boundary of f, or the following conditions are satisfied:

1. $e_1, e_2, \ldots, e_j \in E(G)$.

2. There is a face of Π with both e_j and v on its boundary.

3. e_1 is on the boundary of f.

4. $e_1^*, e_2^*, \ldots, e_j^*$ is a path in the dual graph Π^*.

Given an insertion path, an edge can be inserted into Π starting from an arbitrary point in face f (consider this a 'dummy vertex' for the moment) and ending at vertex v in such a way that it crosses precisely the edges $e_1, e_2, \ldots, e_j$.

Then, suppose we have a collection of insertion paths $p_1, p_2, \ldots, p_\ell$ whose associated end vertices are $v_1, v_2, \ldots, v_\ell$. If they can all be inserted into Π in the above fashion, such that they are pairwise non-crossing, then we say that they collectively constitute a *star insertion path*. By inserting a dummy vertex z into face f and attaching the beginnings of each insertion path to z, the star comprising of z and the edges $\{(z, v_i) \mid i = 1, 2, \ldots, \ell\}$ can be inserted into Π in such a way that they cross precisely the edges in $p_1, p_2, \ldots, p_\ell$. For a fixed face f, and a fixed set of end vertices $S = \{v_1, v_2, \ldots, v_\ell\}$, we say that a star insertion path which crosses the fewest edges with respect to all possible star insertion paths into f with the end vertices S, is a *crossing minimal star insertion path* for f and S. Figure 2.1 displays an example of insertion paths forming a star insertion path. However, the example is not a crossing minimal star insertion path, because clearly edge (z, v_3) can reach face f with fewer crossings than in Figure 2.1.

At each iteration we begin with a combinatorial embedding Π corresponding to some drawing of G. The processes within an iteration are summarised in the following procedure, for a given vertex $v \in V(G)$:

Procedure 1:

P1: Compute the reduced combinatorial embedding $\Pi - v$.

P2: Intelligently (see Procedure 1.1) choose a face f of $\Pi - v$. Compute the number of crossings resulting from a crossing minimal star insertion path into face f for the star comprising of v and its incident edges.

P3: If the total number of crossings has reduced, then insert the star comprised of v and its incident edges into f according to a crossing minimal star insertion path.

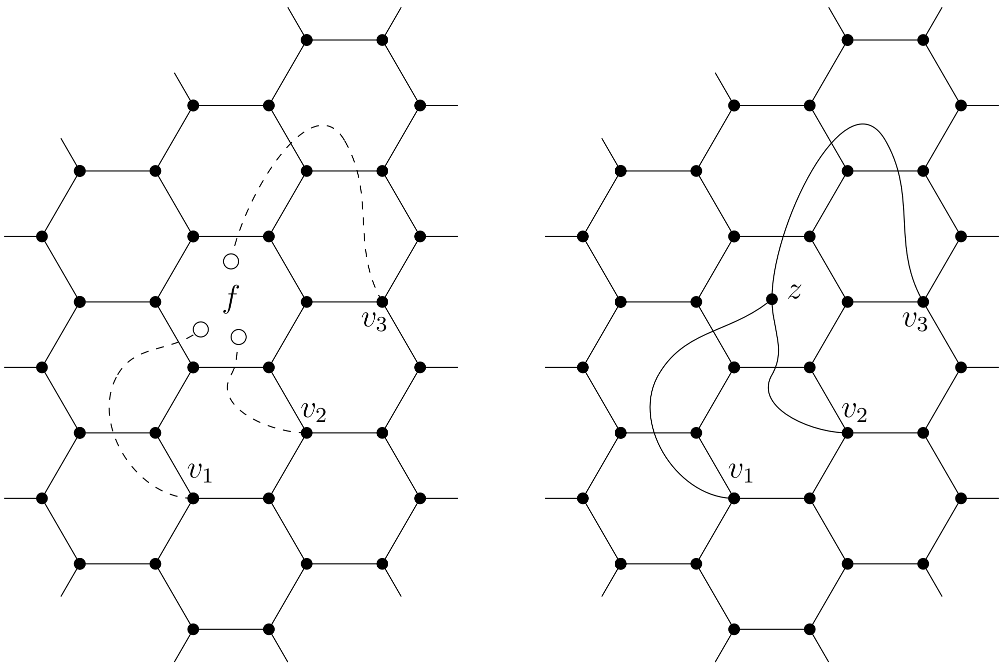


Figure 2.1: On the left, a collection of insertion paths forming a star insertion path for face f and vertices v_1, v_2, v_3. On the right, the star comprising of vertex z and the edges $\{(z, v_1), (z, v_2), (z, v_3)\}$ has been inserted into face f.

P4: Replace each introduced crossing with a dummy vertex of degree 4, and obtain a new combinatorial embedding. Call this new embedding Π and begin the next iteration.

Note that Step P2 is equivalent to solving the fixed embedding star insertion problem for the vertex v (and its incident edges) in $\Pi - v$. To achieve this, we use the algorithm described in Chimani et al. [34] on page 376. Since this is an important step in our heuristic, we include its description here. We begin by utilising a simple merging procedure in the dual graph of $\Pi - v$. For each vertex $w \in N_G(v)$, we perform the following steps:

Procedure 1.1:

1. Let C be the cycle, in the dual graph, formed by the dual vertices of those faces that are incident to w. Contract C into a single vertex, called d_w, and retain the same indices for the dual vertices which were

not contracted (see Figure 2.2 for an example.). Remove any resulting multi-edges.

2. Find shortest paths in the dual graph with d_w as the source.

3. Store the distance from d_w to each other dual vertex (those which were not contracted), and for those dual vertices that were contracted in step 1, set their distance to zero.

4. Discard changes to the dual graph so that the above steps can be repeated with a different neighbour of v.

After each w in above procedure, we have a shortest distance from each dual vertex to a face which is incident to w. Then, after all $w \in N_G(v)$ have been considered, the dual vertex possessing the minimum sum of distances corresponds to the optimal face, say face f, for the new placement of v. Therefore, we have found a crossing minimal star insertion path for f and $N(v)$. The corresponding insertion paths can be determined from the shortest path trees.

2.3 Design methodology

In this section we outline some of the design choices and data structures of the highly practical implementation which is used for the experiments described in Section 2.5. We call this implementation Quickcross.

2.3.1 Initial embedding schemes

Since we focus on a fixed embedding at each iteration, the initial combinatorial embedding of G obviously plays a significant role in the performance of the heuristic. Any drawing method can be used to compute an initial embedding, but here we discuss just three possibilities. The first method

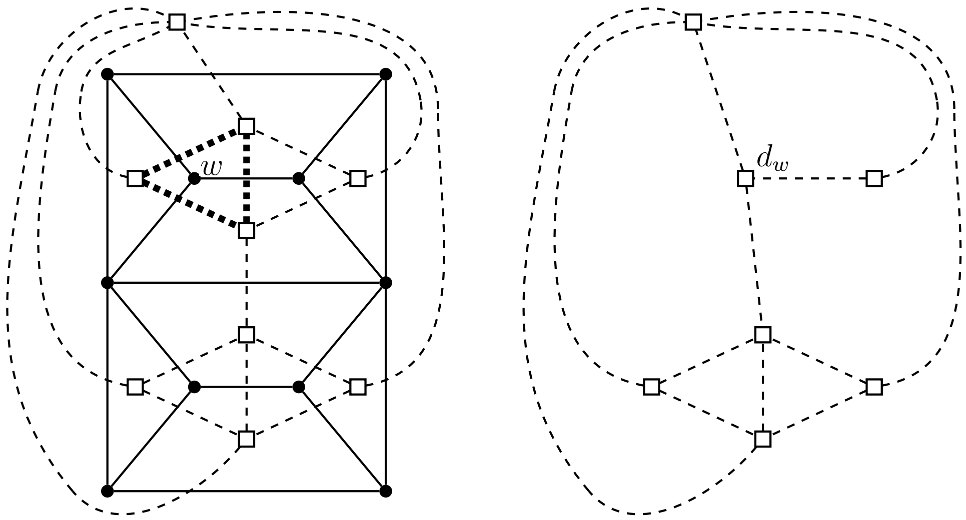


Figure 2.2: The contraction operation in the dual graph for the vertex $w \in N_G(v)$. Dual edges are dashed and dual vertices are squares. The thick dashed edges form the cycle C around w which is contracted and then multi-edges are discarded. The resulting 'merged' dual vertex is d_w.

produces an embedding quickly, however the initial number of crossings can be as large as $\binom{n}{4}$. The second method is slower to compute but the initial number of crossings is usually much smaller for the case of sparse graphs. The third method is an implementation of a force-directed graph drawing algorithm. We will refer to these three initial embedding schemes as *circle*, *planar* and *spring*, respectively.

Circle embedding scheme

The first initial embedding scheme, which we call "*circle*", produces an embedding using the following procedure. We first assign each vertex a coordinate on the unit circle. Specifically, we place each vertex $i = 1, ..., n$ at coordinate $(\cos(2i\pi/n), \sin(2i\pi/n))$. Then, the edges are drawn as straight lines, and the crossings can be easily computed. An upper bound on the number of crossings for a drawing obtained by this method can be seen by following a simple counting argument. In the case of complete graphs, the circle embedding in fact produces a drawing of K_n with number of crossings equal to the maximal crossing number [122].

Lemma 2.7. *The maximum number of crossings in a drawing obtained by the circle embedding scheme is* $\binom{n}{4} = \frac{1}{24}(n^4 - 6n^3 + 11n^2 - 6n)$.

Proof. The maximum number of crossings is attained by the complete graph K_n. In K_n, label the vertices from 1 to n in a clockwise fashion, then any set of 4 vertices $\{a, b, c, d\}$, where $a < b < c < d$, corresponds to exactly one crossing involving the edges (a, c) and (b, d). Thus the total number of crossings is $\binom{n}{4}$. □

Planarisation based embedding scheme

The second initial embedding scheme, which we call "*planar*", utilises a sequence of solutions to the star insertion problem. This idea has been considered as a heuristic for crossing minimisation in its own right (e.g. see Chimani et al. [34]), and involves constructing an embedding in a way which is similar to the planarisation method. We begin by finding any chordless cycle of G (if none exist then G is acyclic and $cr(G) = 0$) along with an embedding Π of this cycle, then iteratively perform the following:

1. Find a vertex $v \in V(G)$ which is not yet in Π, and such that there exists at least one edge in $E(G)$ which connects v to a vertex already present in Π. Denote by F the set of all edges between v and any vertices already present in Π.

2. Find a face f of Π such that a crossing minimal star insertion path, into f, of the star comprising of v and the edges in F, introduces the least number of crossings among all faces of Π.

3. Insert, into f, the star comprising of v and the edges in F according to a crossing minimal star insertion path.

4. Replace each introduced crossing with a dummy vertex of degree 4 to obtain a planar graph, and compute a new combinatorial embedding. Call this new embedding Π and begin the next iteration.

At each step of the procedure we are building upon the embedding, one vertex at a time, until we have an embedding corresponding to some drawing of the full graph G. As will be demonstrated in Section 2.5, this method, although still computationally efficient, is in practice slower than the circle embedding, particularly for dense instances. However, in Section 2.5 it will also be seen that this method tends to result in an initial embedding with many fewer crossings, and hence substantial processing time is saved in the subsequent iterations of the main heuristic. For this reason, this is the default embedding choice in our implementation of the heuristic.

Spring model embedding scheme

The third initial embedding scheme, which we call "*spring*", comes from the area of force-directed graph drawing. In [87] Kamada and Kawai describe a method for drawing a graph which minimises the energy of a spring model representation of the graph. The resulting number of edge crossings is not taken into consideration in the spring model, however, especially for the case of sparse graphs, it will be demonstrated in Section 2.5 that the resulting drawings often provide an initial embedding with relatively few crossings. Of course, there are other force-directed graph drawing algorithms which could be used (e.g. see [50, 73]) and we make no claim here that [87] is the best for use in our heuristic.

It should be noted that, technically, any combinatorial embedding corresponding to a valid drawing of G can serve as an initial embedding. Indeed, our heuristic could be applied as a post-processing step of the planarisation method, or any other similar heuristic which results in a valid drawing. To accommodate this, we have included in our implementation an option for user to specify their own initial combinatorial embedding, or to provide vertex coordinates for a straight-line drawing obtained from any drawing routine.

2.3.2 Minimisation schemes

The heuristic descends towards its solution by selecting vertices for reinsertion and identifying if they can be reinserted in order to reduce the current number of crossings. We call the method in which the heuristic descends towards its solution a "minimisation scheme" and in the below we discuss three implemented minimisation schemes.

The first minimisation scheme, which we call "*first*" works as follows. We consider vertices one at a time, in the order of their labels. In the first iteration, the first vertex considered is the one with the smallest label, and in subsequent iterations the first vertex considered is the one that follows the vertex that was re-inserted in the previous iteration. Plainly, a vertex which was just reinserted in the prior iteration does not need to be considered again in the current iteration because it was already inserted according to a crossing minimal star insertion path and the embedding has not changed since. As soon as a vertex is found which can be re-inserted in such a way that the number of crossings is reduced, we fix this improved position and begin the next iteration.

The second minimisation scheme, which we call "*best*", works as follows. We consider each of the vertices, and determine which should be re-inserted so as to gain the greatest reductions in crossings. To achieve this, similarly to "*first*" we consider vertices one at a time, in the order of their labels. Then, in subsequent iterations, the first vertex considered is the one that follows the vertex that was re-inserted in the previous iteration. In the case of a tie between two vertices providing the same reduction in crossings, we simply choose the vertex which was considered earliest (although more involved strategies could be devised). Then, we fix the improved position of that vertex and begin the next iteration.

The third minimisation scheme, which we call "*biggest face*", comes from an observation made during experimentation; re-inserting a vertex v into the face of $\Pi - v$ with the most edges (the 'biggest face') often provides an improvement. Intuitively this makes sense as the biggest face is 'close' to a relatively large number of vertices. This scheme allows for a significant speed increase during the early iterations because we may assume that the vertex can be placed in the biggest face and then find the shortest paths only once, using the dual vertex corresponding to the biggest face as the source, as opposed to the other schemes which require shortest paths to be computed up to $\Delta(G)$ times. As will be shown in Section 2.4, computing the shortest paths is the most time-consuming process in our heuristic and hence for dense graphs, where $\Delta(G) = \Theta(n)$, we gain a significant speed increase. If the biggest face does not provide an improvement, other faces can then be checked according to one of the other minimisation schemes. In our implementation, if it happens that the biggest face does not provide an improvement for a certain number (specified by the user) of consecutive iterations, we stop checking the biggest face first and instead continue with the "*first*" minimisation scheme from that point forward.

2.3.3 Efficiently handling the dual graphs

In each iteration, and for each vertex considered, the steps of the heuristic require the dual graph of the current embedding minus one vertex. It is possible that we may need to consider many or even all of the vertices, particularly if we use the "best" minimisation scheme. Since the dual graph is likely to be quite similar for each removed vertex, it is undesirable to construct it from scratch each time. Instead we use a simple updating procedure to avoid this. We compute the dual graph once per iteration, with all vertices present. Then, each time a vertex (along with its incident edges) is deleted from G, the result in the embedding is that some pairs of faces (on either side of the planarised edges being deleted) are merged into single faces. In the computed dual graph, this corresponds to contracting the dual edge connecting the two faces on either side of each of these planarised edges (see Figure 2.3). Recall that each edge of the embedding corresponds precisely to an edge of the dual graph. We keep these edge indices consistent in our implementation to help simplify the above process.

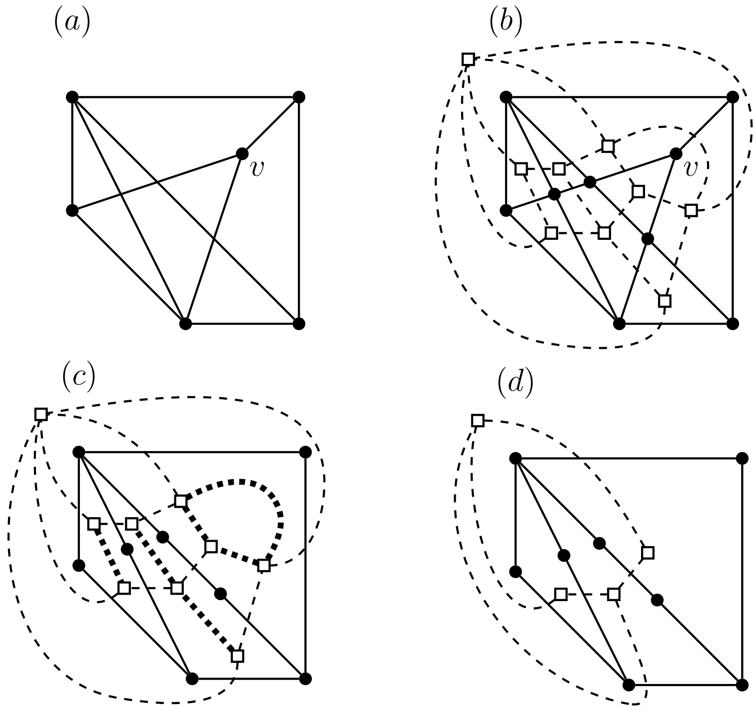


Figure 2.3: Dual edges are dashed and dual vertices are squares. In (a), vertex v is to be deleted. The current embedding and its corresponding dual graph are shown in (b). In (c), v has now been deleted and the thick dashed edges are the dual edges which are to be contracted. Then (d) shows the result after multi-edges have been discarded.

2.3.4 Pre- and post-processing schemes

Pre-processing schemes for crossing number heuristics are well understood and are reported on in [66] and [31]. We briefly outline the usual pre-processing schemes. The crossing number of a disconnected graph is the sum of crossing numbers of each of its connected components. Similarly, the crossing number of a 1-connected graph is the sum of crossing numbers of its maximal bi-connected components. Therefore, we can decompose any input graph into its maximal biconnected components (also known as blocks) and handle them individually. Maximal biconnected components of a graph can be identified in $O(n + m)$ time by using a slight modification to a depth first search algorithm (see [63] page 52). We make use of this pre-processing step in our implementation of Quickcross. One important benefit is that this allows us to assume that any graph submitted to our heuristic is biconnected, and hence we can assume that any graph with one vertex removed is connected.

We briefly summarise some of the effective post-processing schemes utilised in [66] and [31]. In [66] it was identified that an effective post-processing strategy for the planarisation method was to delete some set of edges from the final drawing and attempt to reinsert them in a different order to produce fewer crossings. This is effective for the planarisation method because when inserting an edge, the optimal insertion path does not include any information about those edges which are not yet inserted. Thus an optimal insertion for one edge may adversely effect the optimal insertions of subsequent edges. The most effective way to perform the above was recommended in [31] to be: after inserting an edge, try to remove and reinsert every other edge already in the graph to produce fewer crossings. For each of these edge insertions, new shortest paths need to be considered, and so this method can become very time consuming. However, in [66] and [31], these strategies achieved significant improvements in the final number of crossings compared to no post-processing for the graph sets used in experimentation. Schemes analogous to these can also be used to improve the solutions from our heuristic, and we investigate this further in Section 2.6.

2.3.5 Data structures

To store a combinatorial embedding Π, a list structure containing the following information is utilised: For each edge $e = (u, v)$, this list stores u and v along with four indices; the edge index of the edge immediately clockwise from e around vertex u, the edge index of the edge immediately anti-clockwise from e around vertex u, and then likewise for vertex v.

Additionally, the following list structures allow for the efficient modifications of the embedding at each iteration. The *crossing order* of an edge $e = (u, v)$ where $u < v$ is a list of the edges which currently cross e in the order starting from the closest crossing to u. Along with the crossing order list, there is the *crossing orientation list*. The crossing orientation is essentially the cyclic order of edges around a dummy vertex in the embedding. Suppose that within the crossing order entries of edge $e_1 = (u_1, v_1)$, we have the entry $e_2 = (u_2, v_2)$ where $u_1 < v_1$ and $u_2 < v_2$. Then the corresponding crossing orientation entry is stored as 1 to indicate that the order of the edges when traversing clockwise around the dummy vertex have the end-vertices u_1, u_2, v_1, v_2, or -1 to indicate that the order is u_1, v_2, v_1, u_2. Note that these are the only two possible orders (see Figure 2.4 for an example).

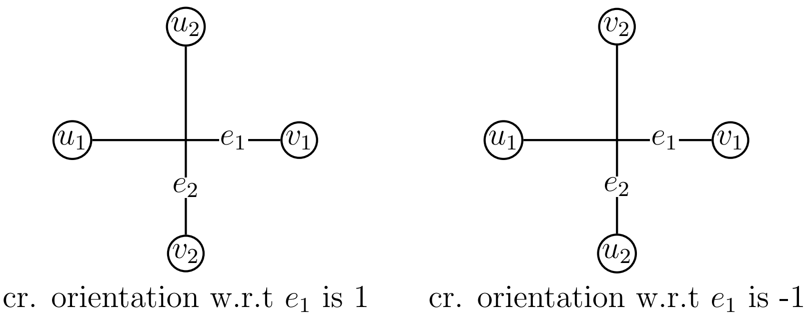


Figure 2.4: If $e_1 = (u_1, v_1)$ crosses $e_2 = (u_2, v_2)$ where $u_1 < v_1$ and $u_2 < v_2$, the two possibilities for the crossing orientation are displayed.

We now provide a simple example of the data structures described above. Consider the graph drawing in Figure 2.5. The embedding corresponding to the drawing is stored as follows. First, the six indices for each edge, as described above, are:

$e_1 : 1, 4, e_1, e_1, e_3, e_5$
$e_2 : 2, 5, e_3, e_3, e_4, e_5$
$e_3 : 2, 4, e_2, e_2, e_5, e_1$
$e_4 : 3, 5, e_4, e_4, e_5, e_2$
$e_5 : 4, 5, e_1, e_3, e_2, e_4$

Then, the crossing order list is:

$e_1 : e_2, e_4$
$e_2 : e_1$
$e_3 : e_4$
$e_4 : e_3, e_1$

Lastly, the crossing orientation list is:

$e_1 : 1, 1$
$e_2 : -1$
$e_3 : 1$
$e_4 : -1, -1$

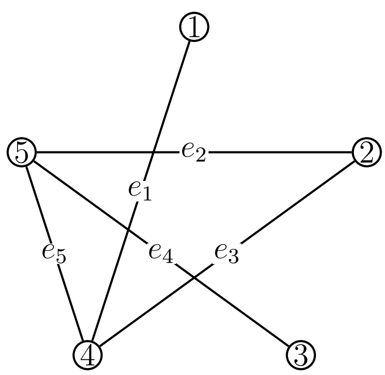


Figure 2.5: A drawing of a graph with vertex and edge indices as labelled.

2.3.6 Difficulties arising from fixed embeddings

During the intermediate steps of Quickcross, we have observed two cases which required specialised techniques to handle. We have determined that these cases were a result of the fixed combinatorial embedding of which Quickcross is based upon. We will now show how to resolve these cases and in doing so, also determine that the final embedding produced by Quickcross corresponds to a good drawing (according to Definition 1.4).

The first case happens when a newly inserted edge crosses another edge more than once. Of course, by Lemma 1.5, it is known that in an optimal drawing this is never the case. However, during an intermediate step of Quickcross, it can arise. Suppose that vertex v has just been reinserted by Quickcross, and consider Figure 2.6. The thick dashed lines represent 'busy' sections of the drawing in which no better insertion path between v and w exists. Therefore the best insertion path for e_1 is to cross e_2 twice. If this happens, e_2 will appear twice on the crossing order list for e_1 and there is not enough information to determine which entry corresponds to which crossing. To avoid this confusion, if edges e_1 and e_2 cross each other more than once, then e_1 is subdivided into a chain of edges such that none of the resulting edges cross e_2 more than once. A check is then performed in future iterations to see if the set of edges resulting from an earlier subdivision still cross any edge more than once. If not, those subdivisions are removed and the edges are merged back into a single edge. Note that subdividing an edge can not change the crossing number of a graph and so this is a safe procedure. Even so, Lemma 2.8 shows that by the time the heuristic concludes, no two edges cross each other more than once and hence, by this time, all previous subdivisions will have been reverted. Note that, in practice, these subdivisions are a rare occurrence.

Lemma 2.8. *In the drawing D corresponding to the final embedding from*

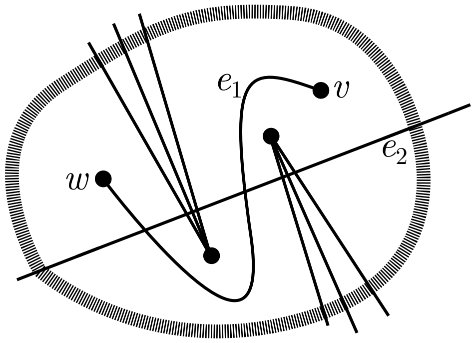


Figure 2.6: A situation in which crossing edge e_2 twice is the optimal insertion path for e_1.

the conclusion of Quickcross, no two edges cross each other more than once.

Proof. Let D have k crossings, and suppose that e_1 crosses e_2 more than once. We will argue that Quickcross is able to find a superior embedding, contradicting the fact that Quickcross has concluded. Let c_1 and c_2 be two crossings of e_1 and e_2 such that traversing along e_1 we arrive first at c_1 and then later at c_2 without further crossing e_2 inbetween. During a single iteration of Quickcross, at least one of e_1 and e_2 may be rerouted in search of a path with fewer crossings. Consider rerouting e_1 to be 'near' e_2 in the way shown in Figure 2.7 (a). In doing so we remove some number of crossings, say d_1, from e_1 and and at least one of c_1 and c_2. Figure 2.8 shows that we cannot assume that both c_1 and c_2 can be removed. However, we may introduce new crossings in this process. In particular, e_1 now crosses those same edges that e_2 crosses between c_1 and c_2, say d_2 in number. Then because Quickcross has concluded, we must have $k - d_1 - 1 + d_2 \geq k$, and hence,

$$d_2 - 1 \geq d_1. \tag{2.1}$$

Similarly we may reroute e_2 to be 'near' e_1 as in Figure 2.7 (b). In doing so we remove d_2 crossings and at least one of c_1 or c_2, and add in d_1 crossings. Then because Quickcross has concluded, we have $k - d_2 - 1 + d_1 \geq k$, and hence,

$$d_1 - 1 \geq d_2. \tag{2.2}$$

Since 2.1 and 2.2 cannot both be true, one must provide an improvement, and since neither of them require us to move any vertices, they can both be performed by Quickcross, contradicting the assumption that Quickcross has concluded. □

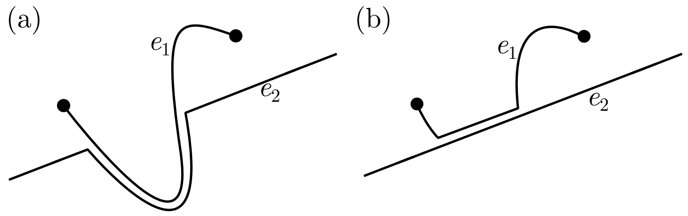


Figure 2.7: Modifications which remove one of the crossings c_1 or c_2 by shifting e_1 to be close to e_2 and vice-versa.

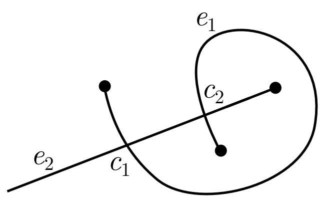


Figure 2.8: Redirecting edges e_1 or e_2 as in Figure 2.7, can only remove 1 of c_1 or c_2.

Next, it may happen that the current embedding possesses a multi-edge between a dummy vertex and one of the original vertices of G. This multi-edge causes undesirable behaviour in several of the procedures of Quickcross. Let edge $e_1 = (v, w)$, then this case can arise when e_1 crosses another edge, say e_2, which is also incident to w. As with the first case, this can happen during the intermediate steps of Quickcross and a situation in which this arises is shown in Figure 2.9 (a) (vertex v has just been reinserted). In the subsequent iterations, if such a crossing becomes the closest crossing to vertex w, for both edges e_1 and e_2, then the embedding contains a multi-edge between the dummy vertex for this crossing and w. Figure 2.9 (b) shows that this may occur when a vertex, which was previously 'blocking' a good v to w path is subsequently moved (vertex y in the figure). We

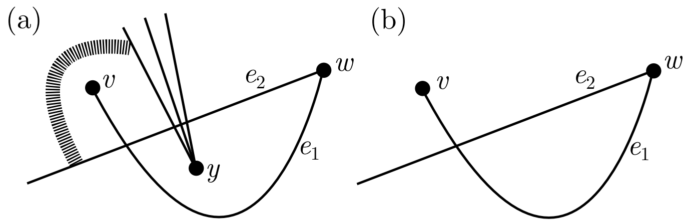


Figure 2.9: In (a), a situation in which crossing edge with a common end-vertex is the optimal insertion path. In (b), if vertex y is subsequently reinserted (somewhere else) then in the embedding, a multi-edge is created.

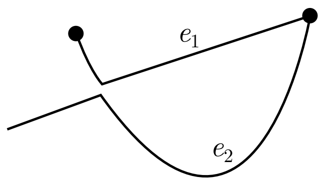


Figure 2.10: Modification which removes a crossing by interchanging a segment of e_1 and e_2.

employ a simple operation, shown in Figure 2.10, which modifies the current embedding, essentially in the same way as in Lemma 1.5, and removes this crossing. Another simple argument shows the following:

Lemma 2.9. *In the drawing D corresponding to the final embedding from the conclusion of Quickcross, no two edges with a common end-vertex cross.*

Proof. The proof is very similar to Lemma 2.8. Let D have k crossings, and e_1 crosses e_2 and vertex v is common to both e_1 and e_2. We will argue that Quickcross is able to find a superior embedding, contradicting the fact that Quickcross has concluded. Let c be the crossing between edges e_1 and e_2 and consider interchanging the segment of e_1 between c and v with the same segment of e_2 as in Figure 2.10. We may do this because we know that e_1 and e_2 are both incident to v. Now we have removed crossing c and e_1 crosses those edges that e_2 crossed in the interchanged segment, and vice-versa. Therefore the total number of crossings has reduced, by one, which is the required contradiction. □

These two cases usually arise when nearby edges have not been moved by Quickcross for some time, and hence are not drawn optimally in the current drawing. The heuristic concludes when it has tried the reinsertion process for every vertex and has been unable to find a drawing with fewer crossings. Lemmas 2.8 and 2.9 lead to the following corollary:

Corollary 2.10. *The final drawing D obtained from Quickcross is a good drawing.*

Proof. Lemmas 2.8 and 2.9 show that conditions 1 and 2 of a good drawing (Definition 1.4) are satisfied at the conclusion of Quickcross, and condition 3 holds trivially. □

2.3.7 Updating the crossing lists

In this subsection we describe how the crossing order and crossing orientation lists are updated in the implementation of Quickcross. Managing these lists effectively is an important ingredient in the efficiency of the overall heuristic.

Crossing order list

At the beginning of an iteration of Quickcross, the crossing order list and crossing orientation list correspond to the dummy vertices of the current embedding Π. Then, after a vertex v is chosen to be re-inserted, these lists must be updated to reflect its new placement. Let f be the face of the reduced embedding $\Pi - v$ into which v will be inserted. Note that f also corresponds to a vertex in the dual graph of $\Pi - v$, which we refer to as 'vertex f'. Similarly, given a vertex y in the dual graph of $\Pi - v$, we refer to the corresponding face as the 'face y'. To insert v into f, we must direct a number of incident edges of v into their new placement, possibly crossing other edges of $\Pi - v$ in doing so. Because any particular edge of $\Pi - v$ may be crossed by many of these incident edges, the order in which the new edges should be listed in the crossing order list of a given existing edge must precisely correspond to the intended new embedding. This is not a trivial procedure and we now describe how to compute this ordering, and illustrate the process with an example.

While finding shortest paths in the dual graph of $\Pi - v$, we calculate a directed shortest path tree with f as the source (step 2 in Procedure 1.1). Denote this tree as T. Note that T may not be a spanning tree of the dual graph because the algorithm terminates once we have found shortest paths to faces incident to each of the neighbours of v. We consider two cases to elucidate the difficulty, and discuss a method to resolve it.

For $u \in N_G(v)$, let $h(u)$ be the vertex in T corresponding to the face which has u on its boundary (the 'last' face on the new insertion path for u and f). Suppose that in T, for all $u \in N(u)$, the f to $h(u)$ paths are pairwise edge disjoint. In this case, there are no two edges, both incident to v, which are required to cross the same edge of $\Pi - v$. An example of this is shown in Figure 2.11. Updating the crossing order list for this case is simple. Suppose that e is an edge which is incident to v, then it is simply a matter of identifying the edges of G which e will be crossing in its new placement, searching the current crossings on that edge, and inserting e into the appropriate place.

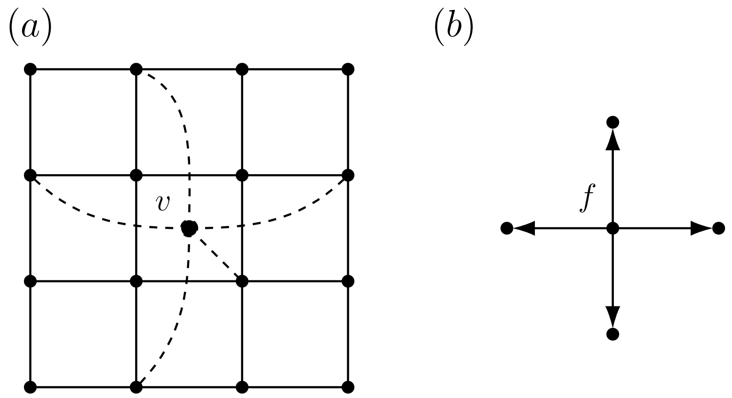


Figure 2.11: In (a), vertex v has been reinserted into face f and its incident edges reinserted. No edge of $\Pi - v$ is crossed by more than one of the edges incident to v. In (b), the corresponding tree T is shown.

The second case becomes more complicated. Suppose that multiple edges, each incident to v, cross the same edge of $\Pi - v$ in their new placement. An example is given in Figure 2.12 (a). The difficulty lies in determining the order in which these edges will cross the existing edge (such as e_1 or e_2 in Figure 2.12 (a)), and hence the order in which they should be inserted into these crossing order lists. To identify the correct order, we use information from $\Pi - v$ along with a post-order tree traversal of T. Note that the embedding $\Pi - v$ also provides an embedding of T, and this embedding is needed to determine the correct order of vertex visits during the tree traversal. During a post-order tree traversal, once a vertex has been visited for the last time, we say that the traversal has *completed* that vertex. Once the post-order traversal has completed a particular vertex of T, we know the full order of crossings on the corresponding edge. In the following example we discuss how to find the ordering for the edges e_1 and e_2 in Figure 2.12 (a). Then, the pseudocode in Algorithm 2.1 gives the required steps for finding these orderings.

Example 2.11. Vertex v has been inserted into face f and each edge incident to v has been inserted, according to shortest paths, as in Figure 2.12 (a). The edges e_1 and e_2 of $\Pi - v$ are crossed respectively by 3 and 2 of the edges

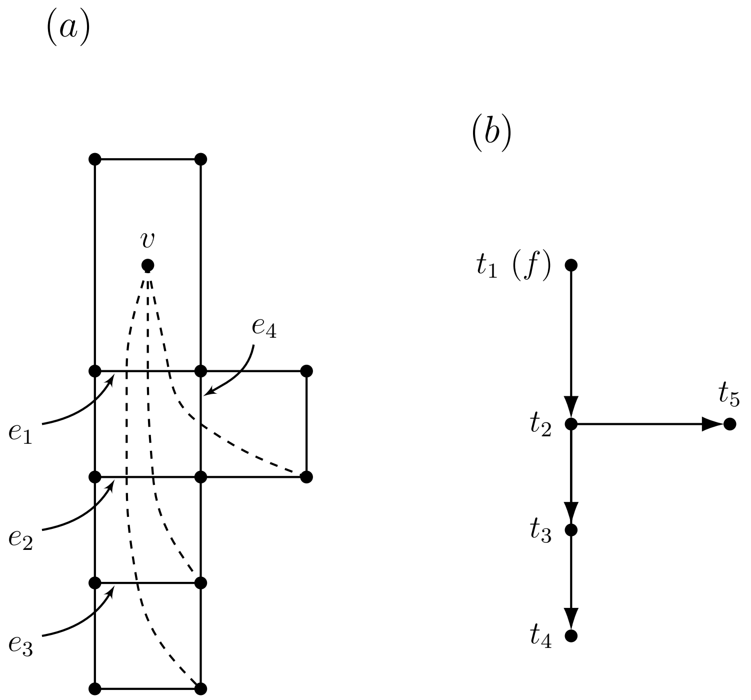


Figure 2.12: In (a), vertex v has been reinserted into face f and its incident edges reinserted. Edges e_1 and e_2 are now crossed by multiple of the edges incident to v. In (b), the tree T corresponding to the drawing (without v) in (a).

incident to v. Our task is to identify the order in which the edges incident to v will be placed into e_1 and e_2's crossing order lists. The corresponding shortest path tree T is shown in Figure 2.12 (b) and we will perform a post-order tree traversal of T. Note that vertex t_1 coincides with face f. Suppose that the post traversal has just completed a vertex y. Assume that we know apriori the crossing order (of the edges incident to v) for all of the edges on face y. Let e_y be the edge in face y, which corresponds to to edge $(\text{parent}(y), y)$ in T. Then by scanning face y in an clockwise manner, beginning at edge e_y, and concatenating the known crossing orders (of the edges incident to v) of all edges on y, we compute the crossing order of edge e_y. Thus by the time the post tree traversal has concluded, the crossing order for all new crossings has been determined. Note that the found order may then need to be reversed, depending on the vertex indices of the planarised edge which is being crossed. We will now perform the above tasks and Figure 2.13 illustrates the known crossing orders at each step.

1. According to the embedding of T determined by $\Pi - v$, the post-order tree traversal first completes vertex t_4 and there is only a single edge incident to v in face t_4 and so no scan is needed.

2. The traversal then completes vertex t_3. Face t_3 contains two of the edges incident to v and so we perform a clockwise scan of face t_3, beginning at edge e_2. The scan reaches vertex w_2, which is the end-vertex of edge (v, w_2) then later we reach edge e_3 which is crossed by edge (v, w_3). Hence we have determined that the crossing order on edge e_2 is to be $\{(v, w_2), (v, w_3)\}$ (which may need reversing depending on the vertex labels of e_2).

3. The traversal then completes vertex t_5 and there is only a single edge incident to v in face t_5 and so no scan is needed.

4. Lastly, the traversal completes vertex t_2. Face t_2 contains three of the edges incident to v and so we perform a clockwise scan of face t_2, beginning at edge e_1. The scan reaches the edge e_4 first, which is crossed only by (v, w_1). The scan then reaches edge e_2, which is crossed by multiple edges incident to v. We already determined edge e_2's crossing order in step 2, and so we use this information. Hence we have determined the crossing order of e_1 to be $\{(v, w_1), (v, w_2), (v, w_3)\}$ (which may need reversing depending on the vertex labels of e_2).

In Algorithm 2.1, we use a list structure C. If an edge e of $\Pi - v$ is crossed by any of the edges incident to v in their new placement, then $C(e)$ will contain the crossing order of those edges which cross e. Thus if a planarised edge e is crossed by a single edge incident to v, then $C(e)$ will just contain a single entry. If a planarised edge e is crossed by many edges incident to v, then $C(e)$ will be an ordered list of each of these edges.

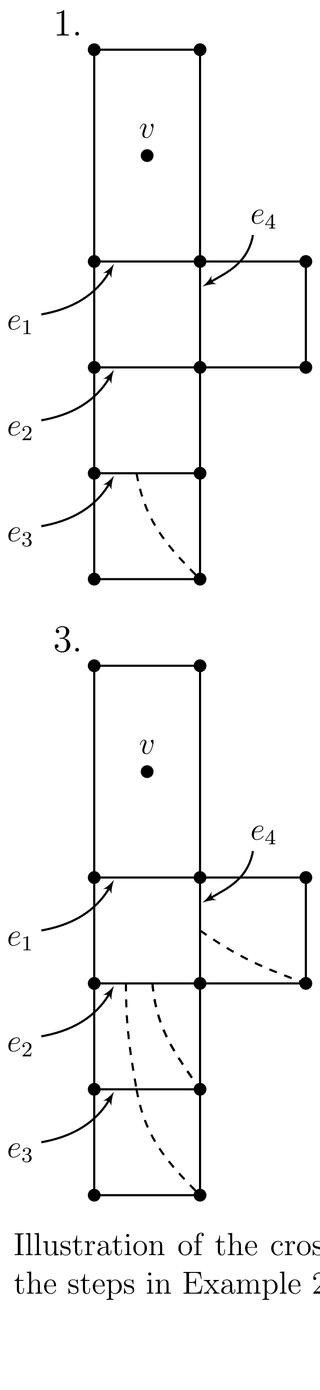


Figure 2.13: Illustration of the crossing orders being determined at each of the steps in Example 2.11.

Algorithm 2.1 For $u \in N_G(v)$, let $h(u)$ be the vertex in T corresponding to the face, incident to u, which will be the 'last' face on the new v to u path. Scanning face f in a clockwise manner involves alternating between checking a vertex of f and then an edge of f. The operation 'conc' is concatenation.

procedure UPDATE CROSSING ORDER(C, T, w, VISITED)
    for $z \in N(w)$ **do**
        if visited$(z) = 0$ **then**
            visited$(z) = 1$
            UPDATE CROSSING ORDER(C, T, z, visited)
    Denote by e_w the edge in $\Pi - v$ corresponding to (parent$(w), w$).
    do clockwise scan of face w beginning at e_w.
    if scan is at vertex u where $u \in N_G(v)$ *and* $w = h(u)$ **then**
        $C(e_w) = \{C(e_w)\}\text{conc}\{(v, u)\}$.
    if scan is at edge $e \neq e_w$ *and* e corresponds to a tree edge. **then**
        $C(e_w) = \{C(e_w)\}\text{conc}\{C(e)\}$.
    return (C, visited)

Crossing orientation list

In order to update the embedding $\Pi - v$ to reflect v's new placement, we must be able to compute the cyclic order of edges around any of the dummy vertices corresponding to crossings. As was discussed in Section 2.3.5, there are only two possible cyclic orderings for these dummy vertices and we called these orderings the *crossing orientation* of the corresponding crossing. If we assume that crossing orientations are currently known at the start of an iteration, then after moving some vertex v into face f, the only change in crossing orientations happens for crossings on edges incident to v. Hence the only crossing orientations which need to be computed within an iteration, are for the new crossings introduced when v is moved into face f. This can be achieved by following a simple procedure when updating $\Pi - v$ to reflect

v's new placement.

Along with the current embedding Π, we have the crossing order list, which lists crossings along any edge $e \in E(G)$ in order, beginning with the crossing closest to the smallest vertex index of e. Let $e = (s,t) \in E(G)$ where $s < t$ and suppose e has d edges crossing it in the drawing corresponding to Π. In Π, e is associated with a chain of edges $e_1, e_2, \ldots, e_{d+1}$ as in Figure 2.14. These new edges may be labelled so that they obey $e_1 < e_2 < \cdots < e_{d+1}$. Now, suppose that v has been reinserted into face f and a number of edges incident to v now cross some edges in $\Pi - v$. Suppose that edge (v,w), where $v < w$, now crosses edge e_k in $\Pi - v$ and we need to compute this crossing orientation. Note that e_k is associated with the original edge $e = (s,t) \in E(G)$ where $s < t$ and the crossing orientation will be with respect to this original edge e. From computing the shortest paths, we know which faces edge (v,w) passes through and the order in which it does so. Let f be the face, incident to e_k, which (v,w) passes through and is closer to v, as in Figure 2.15 (a). Inside face f, we may traverse clockwise one edge away from e_k, say to e_r and observe the end-vertex which is common to both e_k and e_r. This vertex may be s or t or a dummy vertex. If the common vertex is s, then because we traversed clockwise and $s < t$, we know the crossing orientation is 1 with respect to e. Figure 2.15 (a) shows this situation. Similarly if this vertex is t, then we know the crossing orientation is -1 with respect to e. If this vertex is a dummy vertex, then this dummy vertex is of degree 4, and so we may traverse this dummy vertex once, from e_r, and we must arrive at another edge which is associated with e. Suppose that we arrive at edge e_q. Finally, because we created the edges along e in order, if $e_q < e_k$, it implies that e_q is closer to s than e_k is. Then, because we traversed clockwise along face f, we know that the crossing orientation must be 1 with respect to e. This situation is displayed in Figure 2.15 (b). Similarly if $e_q > e_r$, we know that the orientation must be -1 with respect to

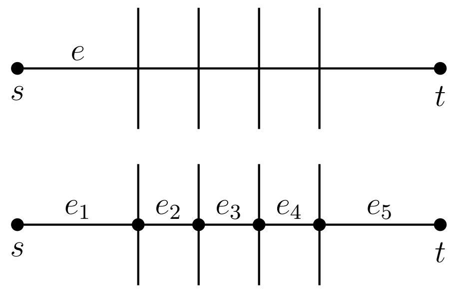


Figure 2.14: $e = (s, t) \in E(G)$ with $s < t$ and 4 crossings and the corresponding chain of edges in Π, where $e_1 < e_2 < e_3 < e_4 < e_5$.

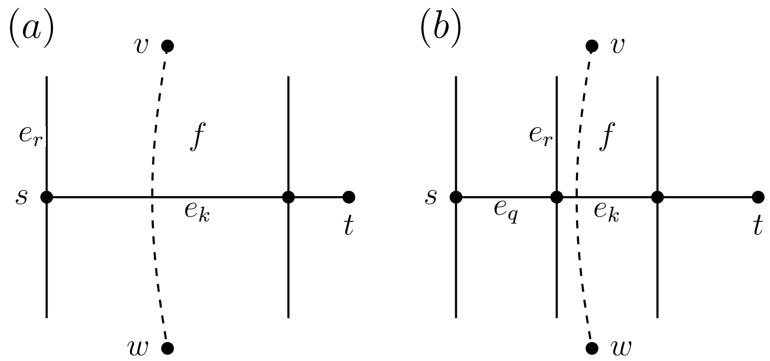


Figure 2.15: Identifying the crossing orientation of a new crossing. Note that edge (v, w) is not a part of face f.

e.

2.4 Implementation and Runtime

We now discuss the runtime of each of the procedures and show that the iterations of the heuristic each run in $O((k+n)m)$ time where k is the number of crossings in the drawing associated with that iteration. Pseudocode for the main loop and two subroutines are displayed in Algorithms 2.2-2.4. The code implements the first minimisation scheme discussed in Section 2.3.2 where an improvement is taken as soon as it is found. There is a level of abstraction left in the pseudocode due to the numerous ways that one could perform the required operations; highly optimised C and MATLAB implementations of the heuristic are available at http://fhcp.edu.au/quickcross. In the discussion below we refer to the pseudocode and summarise the methods used in our implementation.

2.4.1 Implementation

First we discuss the operations involved in Algorithms 2.2-2.4. During the MAIN LOOP procedure, we remove vertex v and its incident edges, which possibly reduces the current number of crossings. Then, after identifying the best possible new placement for v using the SIP(G, Π^*, v) procedure, we have a new number of crossings for v's potential placement and this number is *new cr*. Hence if *new cr* $<$ *current cr* then we have found a drawing with fewer crossings. Once an improvement has been found, Π is updated to reflect the new placement and this involves updating each of the data structures discussed in Sections 2.3.5 and 2.3.6. If an edge is drawn such that it crosses some other edge multiple times, then we subdivide that edge to avoid confusion in the data structures as also discussed in Section 2.3.5. Similarly, if a set of edges resulting from an earlier subdivision no longer crosses any edge multiple times, then the previous subdivisions are reverted.

From the current combinatorial embedding Π, we compute the dual graph Π_0^* (which is then copied into Π^* for modifications). Step P1 of Procedure 1asks us to compute the reduced combinatorial embedding $\Pi - v$. This corresponds to removing v from Π and a set of planarised edges. Because at this stage, it is unknown if the embedding $\Pi - v$ will be utilised for the next iteration, it is quicker to instead modify the dual graph Π^* to reflect the removal of v. This process is done inside of the procedure REMOVE(G, Π^*, v) according to the discussion on contractions in the dual graph in Section 2.3.3. Later, if $\Pi - v$ will be utilised for the next iteration, then it is computed, along with the new placement of v.

The procedure SIP(G, Π^*, v) solves the fixed embedding star insertion problem for the vertex v (into $\Pi - v$). The contractions in Π^*, discussed in Section 2.2, reduce the number of times that shortest paths need to be computed, which is the most costly process of the heuristic. Then the optimal placement for v is given by *newface*, and shortest paths are computed once more with *newface* as the source vertex. The list *shortest paths* stores the tree paths from *newface* to each $w \in N_G(v)$.

Algorithm 2.2 Main procedure of the heuristic. Inputs are a combinatorial embedding Π corresponding to some initial drawing D of G, which is represented by the data structures discussed in Section 2.3.5.

```
procedure MAIN LOOP
    current cr ← $cr_D(G)$
    while true do
        improvement found ← false
        Find the faces of $\Pi$.
        $\Pi_0^*$ ← dual graph of $\Pi$
        for $v \in V(G)$ do
            $\Pi^*$ ← $\Pi_0^*$ (make a copy of $\Pi_0^*$)
            $\Pi^*$ ← REMOVE($G, \Pi^*, v$)
            (new cr, newface, shortest paths) ← SIP($G, \Pi^*, v$)
            if new cr < current cr then
                improvement found ← true
                break
        if improvement found then
            Update $\Pi$ to reflect new placement using newface and shortest paths.
            Check if any subdivisions are needed.
            Check if any previous subdivisions can be removed.
            current cr ← new cr
            continue
        else
            break
    return (current cr, $\Pi$)
```

Algorithm 2.3 Vertex deletion procedure. Given a dual graph Π^* and a vertex v of G, this performs edge contractions in the dual according to the discussion in Section 2.3.3.

```
procedure REMOVE($G, \Pi^*, v$)
    for $e^* \in E(\Pi^*)$ do
        if $e^*$ corresponds to an edge of $G$ which is incident to $v$ then
            Contract $e^*$.
    return ($\Pi^*$)
```

Algorithm 2.4 Star insertion problem solver. Given a dual graph Π^* along with a vertex v of G, this performs edge contractions in the dual graph according to the discussion in Section 2.2. Then the fixed embedding version of the star insertion problem is solved for v.

procedure SIP(G, Π^*, v)
    for $w \in N_G(v)$ **do**
        $\Pi^{**} \leftarrow \Pi^*$ (make a copy of Π^*)
        In Π^{**}, contract the cycle formed by dual edges corresponding to edges incident to w in Π, call the contracted vertex w_d.
        $\text{dist}_w \leftarrow$ Shortest path algorithm(Π^{**}, w_d).
        Set the dist of vertices contracted to form w_d to zero.
    newface $\leftarrow \text{argmin}_k(\sum_{w \in N(v)} \text{dist}_w(k))$
    shortest paths $\leftarrow$ Shortest path algorithm(Π^*,*newface*)
    return (*new cr*, *newface*, *shortest paths*)

2.4.2 Runtime

In this subsection we work through the lines of the MAIN LOOP pseudocode and discuss the time complexity of each operation. The majority of the work is simple vector manipulation and so some detail is left out here. As will be seen, in each iteration, the runtime is dominated by the task of finding the many shortest paths in order to solve the star insertion problem. After our discussion, we conclude that for practical purposes, the steps performed take no more than $O((k+n)m)$ time.

At lines 5 and 6 of the MAIN LOOP we find the faces and dual graph of Π. This can be achieved by scanning the edges of Π in a clockwise manner and time required for this is $O(k+m)$.

Next, during the loop at line 7, we delete a vertex v and search for a better placement for v. Potentially every vertex may be tried before the algorithm moves on. So the procedures inside this loop may be repeated up to n times per iteration.

Inside the procedure REMOVE, which is entered at line 9 of 2.2, a number of edge contractions are performed. In the drawing of G which is associated with the current embedding Π, let k_v denote the number of crossings on the edges incident to vertex v. Then the time required for the corresponding edge contractions is $O(k_v + d(v))$ for each v. Summing over all n vertices in the aforementioned loop, this becomes a worst case of $O(k + m)$.

In the procedure SIP, the contractions at line 4 can be performed in $O(\sum_{w \in N(v)} d(w))$ time, and summing over all vertices, this becomes $O(nm)$. At line 5 we find shortest paths on an unweighted planar graph (a simple breadth-first search) which can be done in $O(k+n)$ time and this is repeated for each $w \in N_G(v)$ by the loop at line 2. Then, summing over all vertices, this becomes $O((k + n)m)$.

Back in the MAIN LOOP the following procedures happen only once an improvement has been found, so only once per iteration. At line 17 we fix the new placement and update the existing data to reflect the new placement. Updating the crossing order list and crossing orientation list as discussed in Section 2.3.6 can be performed in $O(k)$ time. Updating the four clockwise and anticlockwise edge indices discussed in Section 2.3.5 can be done in $O(k+m)$ time.

Any required subdivisions are checked for at line 18 by scanning the crossings on every edge to check whether it crosses the same edge more than once. This scan can be performed in $O(k)$ time. If a subdivision is required then the corresponding lists need to be updated and this also happens in $O(k)$ time. Note that these subdivisions are a very rare occurrence in practice and when they do occur, a check is put in place at each iteration thereafter to see if the subdivision can be undone. This additional check can be performed in $O(k)$ time. If a subdivision is required to be undone, the corresponding lists need to be updated and this happens in $O(k + m)$ time.

We remark that any subdivisions do have an effect on the runtime of future iterations because they cause n to grow, and bounding the time increase is difficult. Because these subdivisions are rare cases which are usually removed swiftly in subsequent iterations, we conclude that for practical purposes the additional runtime is negligible.

The above discussions have considered all major tasks in our implementation and thus we conclude that each iteration can effectively be performed in $O((k+n)m)$ time. Finally, we remark that the total number of iterations is at most the number of crossings in the initial drawing of G. Hence a naïve bound on the total runtime is $O((\bar{k}+n)\bar{k}m)$ where $\bar{k}$ is the initial number of crossings. This emphasises the dependency between the quality of the initial drawing and the overall performance of the heuristic.

2.5 Experiments

2.5.1 Experimental setup

In this section, we consider the performance of our proposed heuristic on various sets of instances. As mentioned previously, we have implemented our heuristic in both C and MATLAB, and here we report on the C implementation, which we call Quickcross.

Each of the experiments reported on in this section were conducted on a 2.6GHz AMD Opteron 6282 SE with 4GB RAM, running Centos 6.7. In order to compare the various schemes discussed in Section 2.3, each experiment is repeated for nine different parameter settings, once for each combination of the three initial embedding schemes (*circle, planar, spring*), and the three minimisation schemes (*first, best, biggest face (bf)*). Then, for each of these nine parameter settings, we use 100 different random permutations of the vertex labels and record the result with the least number of crossings. In such a case, we shall say that the graph was *run with 100 random permutations*.

Where possible, we also make comparisons with the state-of-the-art crossing minimisation heuristics included in the Open Graph Drawing Framework (OGDF) [32]. Included in OGDF are implementations of the planarisation method based on both the fixed and variable edge insertion problem as well as a practical implementation of the approximation algorithm based on multiple edge insertion from [35]. Guided by the experimental results in [31], we attempt to make our comparisons as fair as possible. We denote the heuristic based upon fixed and variable edge insertion as *fix* and *var* respectively and the multiple edge insertion heuristic as *multi*. To initialise the planarisation method, a maximal planar subgraph is computed using OGDF's PQ-tree based planar subgraph algorithm, where the best found solution out of 64 restarts is chosen. We consider four of the post-processing strategies investigated in [31], which include:

- No post-processing, denoted as *none*.
- The edge reinsertion strategy - after all edges are present in the graph, the heuristic deletes each edge, one at a time, and reinserts it, possibly with fewer crossings. This strategy is denoted as *all*.
- The incremental strategy - after each individual edge insertion during the main heuristic, every edge currently present in the graph is deleted, one at a time, and reinserted, potentially with fewer crossings. This strategy is denoted *inc*.
- The incremental (only inserted) strategy - after each individual edge insertion during the main heuristic, every edge except those from the original maximal planar subgraph is deleted, one at a time, and reinserted, potentially with fewer crossings. This strategy is denoted *inc/ins*.

Similarly to our experiments with Quickcross, when each instance is run on OGDF for the above schemes, we use 100 different random permutations of the vertex and edge labels and record the result with the least number of crossings. We also note that the strategies *inc* and *inc/ins* perform a significant amount of additional work; indeed these schemes become the dominant part of the whole algorithm. It was noted in [31] that these incremental schemes produce superior results at a cost of significantly higher runtimes. We observe that the high runtimes are further exacerbated on dense graphs.

We will consider six sets of instances, the first four of which contain sparse graphs, and the latter two of which contain dense graphs. In particular, the sparse instances considered are the sets of instances which were used for benchmarking crossing minimisation heuristics in [31], [65] and [66]. They are known respectively as the KnownCR graphs, the Rome graphs, the AT&T graphs and the ISCA graphs. The dense instances considered are sets of complete graphs, and complete bipartite graphs. We now briefly describe the experiments that will be carried out for each of the sets.

- **KnownCR graphs** - these are a set of instances containing between 9 and 250 vertices, first collected by Gutwenger [65], which can be further partitioned into four families of graphs as follows:
  - $C_i \square C_j$: the Cartesian product of the cycle on i vertices with the cycle on j vertices. These instances contain graphs with $3 \leq i \leq 7$ and $j \geq i$ such that $ij \leq 250$.
  - $G_i \square P_j$: the Cartesian product of the path on $j+1$ vertices with one of the 21 non-isomorphic connected graphs on 5 vertices, where i denotes which of the 21. These instances contain graphs with $3 \leq j \leq 49$.
  - $G_i \square C_j$: the Cartesian product of the cycle on j vertices with one of the 21 non-isomorphic connected graphs on 5 vertices, where i denotes which of the 21. The crossing number of these graphs are only known for some of the G_i and only these cases are included. These instances contain graphs with $3 \leq j \leq 50$.

– The generalised Petersen graphs $P(j, 2)$ and $P(j, 3)$, on $2j$ vertices. We shall only use those of type $P(j, 3)$, as $P(j, 2)$ are easy for heuristics to solve as has already been observed in [31]. These instances contain graphs with $9 \leq j \leq 125$.

Unlike the other sets of instances in this section, all of the crossing numbers for the KnownCR instances are known, and hence we can compare how close the results we obtain for various scheme combinations are to the correct values. In particular, we report on the average relative deviation between the crossing numbers and the values obtained by each heuristic and scheme combination. For these results with Quickcross, we also illustrate the work remaining to be performed during the main loop, by reporting the average relative deviation after only the initial embedding is finished. We also compare the runtimes of the various scheme combinations, separated into the time spent producing the initial embedding, and the time spent in the main loop of the heuristic. Finally, we compare the results between the different scheme combinations of both Quickcross and OGDF.

- **Rome graphs** - these are a set of 11,528 graphs which have been constructed from real-life applications, first described by Di Battista et al. [46]. They contain between 10 and 100 vertices, and are very sparse with average edge density of 1.35. The larger graphs in this set have unknown crossing numbers, since they are too large for the current exact methods to solve. Hence, it is impossible to report on how close QuickCross gets to the true crossing number. However, in [66] and [65], the largest graphs in the Rome set were considered, that is, the 140 graphs with exactly 100 vertices. For these graphs, the average numbers of crossings found for various crossing minimisation heuristics were reported. We repeat this same experiment and compare the results for both QuickCross and OGDF, and also report on the runtimes for each of the scheme combinations. For Quickcross, the runtimes are separated into time spent producing the initial embedding, and time spent in the main loop of the heuristic.

- **AT&T graphs** - these are a set of 311 graphs with between 25 and 312 vertices. Their crossing numbers are not known, and hence we are unable to report on how close our results are to the crossing number. In [31], various crossing minimisation heuristics were compared in the following way. First, each graph was submitted to each heuristic, and the best solutions found overall were recorded. Then, the average relative difference compared to the best found solution was reported, with the instances partitioned according to the number of crossings found in the best solution. Since the best number of crossings found was not explicitly given in [31], we are unable to compare our results to this. Instead, we perform a simpler experiment whose results can be compared in the future. We report the average numbers of crossings found over all 311 graphs, for each different combination of the heuristics and schemes of both Quickcross and OGDF.

- **ISCA graphs** - these are a set of 20 graphs with between 25 and 233 vertices. They began as multigraphs from the ISCA 1985 benchmark set which were then appropriately modified into simple undirected graphs in [31]. Their crossing numbers are not known, however, since there are only 20 graphs, it is possible to compare the performance of each individual graph for each combination of heuristics and schemes of both Quickcross and OGDF.

- **Complete graphs** - Although the crossing number of the complete graph K_n is not known for $n \geq 13$, the value is conjectured, and this conjecture is typically assumed to be correct. For each different combination of heuristics and schemes of both Quickcross and OGDF, we report on how close they get to the conjectured value for complete graphs with between 20 and 50 vertices.

- **Complete bipartite graphs** - Much like the complete graphs, the crossing number of the complete bipartite graph K_{n_1,n_2} is only known in general for $n_1 \leq 6$, but the value is conjectured and typically assumed to be correct. Again, for each different combination of heuristics and schemes of both Quickcross and OGDF, we report on how close they get to the conjectured value for graphs with n_1 and n_2 between 20 and 40.

Note that in the KnownCR and the Rome graphs, we consider individually the performance of Quickcross in the initial embedding, and the main loop of the heuristic. Since we found that the behaviour exhibited is consistent irrespective of the graph set considered, we do not display this breakdown for the subsequent four sets.

2.5.2 KnownCR graphs

We partitioned the graphs into the four families described above and ran each with the different possible combinations of heuristics and schemes. Each graph was run with 100 random permutations and the minimum found solution was compared to the actual crossing number by computing the *percent relative deviation*. Let k denote the minimum found solution, then the percent relative deviation from $cr(G)$ is: $100(k - cr(G))/cr(G)$. The average of these numbers was then taken over each of the four families of graphs and these results are displayed in Table 2.16, which we now describe in detail.

We observe that for the graphs of type $G_i \square C_j$ and $C_i \square C_j$, the *circle* and *planar* embeddings perform very well and they outperform the *spring* embedding, as well as the OGDF *inc/ins* and *none* results by approximately 2.5% or more. However, the OGDF *inc* and *all* schemes are superior for these graphs, in some cases, reaching the optimal solution for every instance. On the other hand, for the graphs of type $G_i \Box P_j$ and P$(j,3)$, the Quickcross *circle* and *planar* embeddings perform relatively poorly while the *spring* embedding performs better. For the $P(j,3)$ graphs the *spring* embedding produced average relative deviations which are approximately equal to the best OGDF results, while they are slightly worse than the best results from OGDF for the $G_i \Box P_j$ graphs. The Quickcross *best* scheme performed worse than *first* and *bf* under the same initial embedding scheme in almost all cases, with the sole exception of the *planar* embedding for $G_i \Box C_j$ and $P(j,3)$.

Final crossings (%) for KnownCR graphs

Schemes	$G_i \Box P_j$	$G_i \Box C_j$	$C_i \Box C_j$	$P(j,3)$
spring,first	3.7560	5.2257	4.9672	3.4101
spring,best	5.5655	6.6789	6.1827	4.8419
spring,bf	3.7016	4.4707	4.5493	3.2199
circle,first	8.7814	2.5407	1.5642	7.8437
circle,best	12.427	6.4100	4.8994	9.9959
circle,bf	7.8796	1.7049	1.7314	6.7071
planar,first	10.321	1.5063	1.6612	7.3148
planar,best	11.372	1.4091	1.7223	7.1430
planar,bf	9.7580	1.4994	1.5837	7.1944
fix,none	16.133	15.672	12.297	29.122
fix,inc	3.6714	1.4214	0.0192	5.4912
fix,inc/ins	12.791	11.552	6.0648	23.161
fix,all	4.5824	1.9308	0.0253	7.0755
var,none	13.786	13.318	10.1881	24.160
var,inc	2.4155	0.5262	0.0000	3.5897
var,inc/ins	10.060	8.9232	4.5974	17.6333
var,all	3.4524	0.8107	0.0000	5.3482
multi,none	14.224	13.998	10.4127	25.253
multi,inc	3.4226	1.2341	0.0232	5.4502
multi,inc/ins	10.717	9.9423	4.5327	20.032
multi,all	3.3576	0.8686	0.0072	4.6817

Table 2.16: Results for the KnownCR graphs run with the different scheme combinations of Quickcross and OGDF. The values are the average percent relative deviation from the crossing number for the four families of graphs within the KnownCR set.

Runtimes were analysed by taking an average over the 100 random permutations for each graph. These times are difficult to display meaningfully in a single figure as there is a large amount of variation within each of the four families of graphs. To that end, Figures 2.17-2.20 display the runtimes for Quickcross on each of the four families, and Figures 2.21 - 2.24 display the equivalent for OGDF. For Quickcross, it can be seen that the *best* minimisation scheme is significantly slower than the alternatives. For OGDF, the *inc* scheme is also significantly slower than the alternatives.

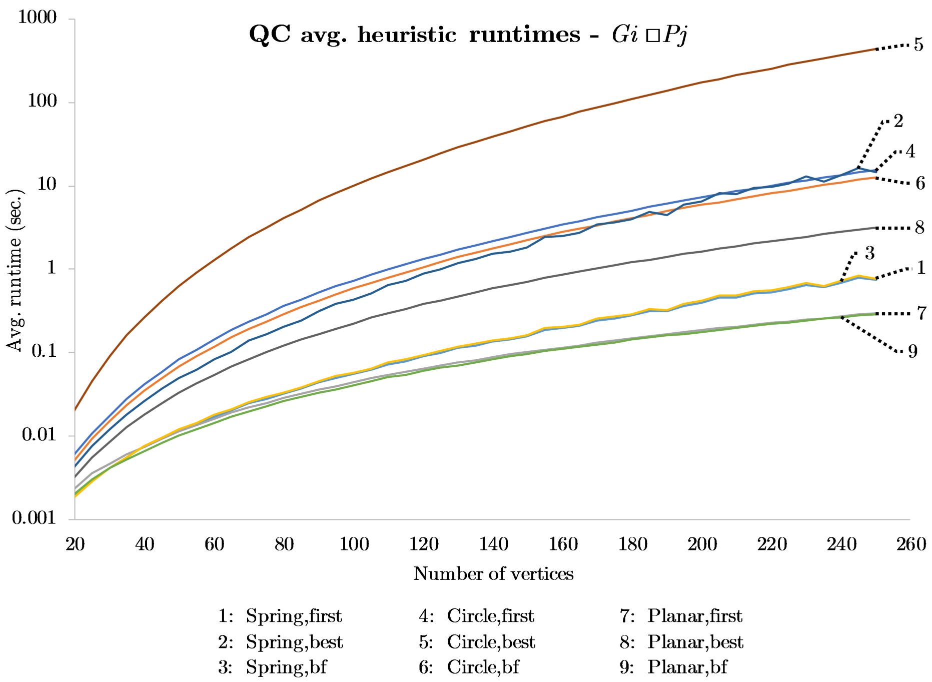


Figure 2.17: Average heuristic runtime of Quickcross per random permutation (sec.) compared to number of vertices for the graphs $G_i \Box P_j$.

We also briefly analyse the initial embeddings of Quickcross for these graphs. In Figure 2.26 we display the average runtimes to complete the initial embedding. We observe that, as indicated in Section 2.3.1, the *circle* embedding computes an initial embedding the quickest, however it creates many additional crossings, seen in Table 2.25, and consequently the full heuristic has a longer runtime. Alternatively, the *planar* embedding scheme computes an embedding almost as quick and the embedding has far fewer crossings,

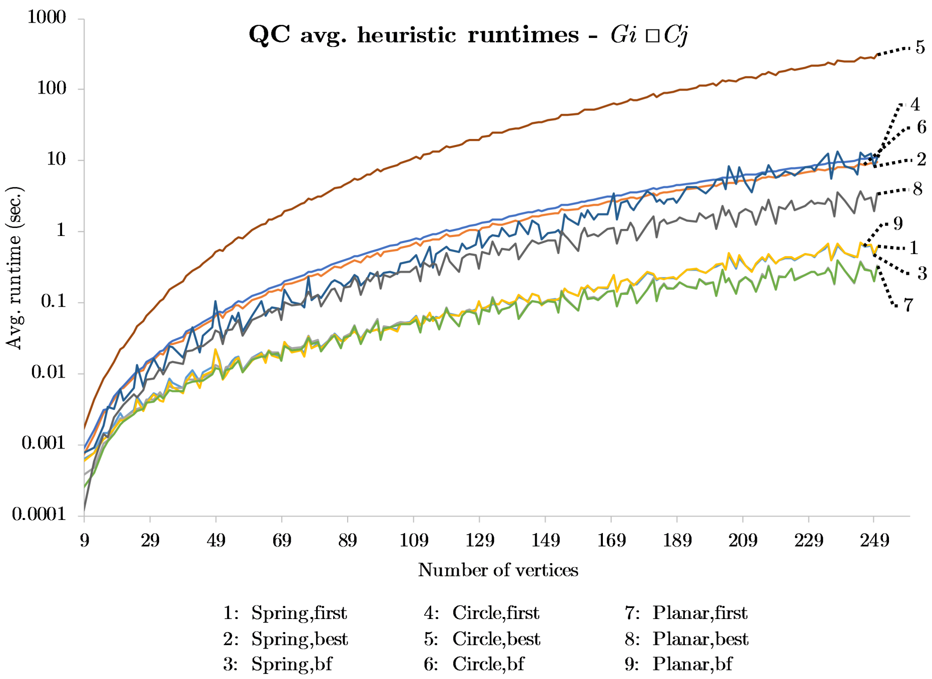


Figure 2.18: Average heuristic runtime of Quickcross per random permutation (sec.) compared to number of vertices for the graphs $G_i \Box C_j$.

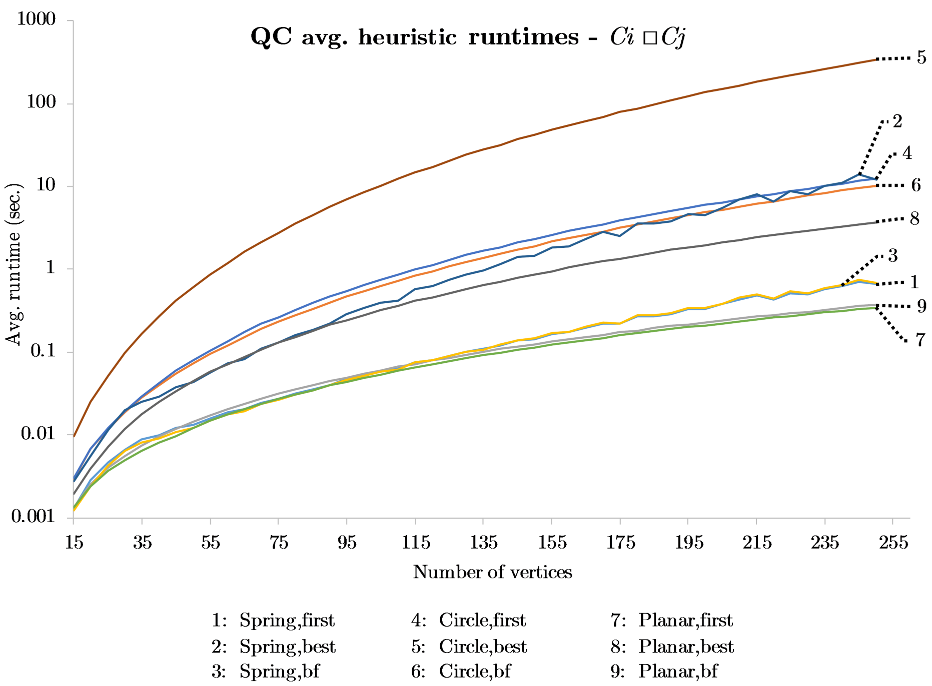


Figure 2.19: Average heuristic runtime of Quickcross per random permutation (sec.) compared to number of vertices for the graphs $C_i \Box C_j$.

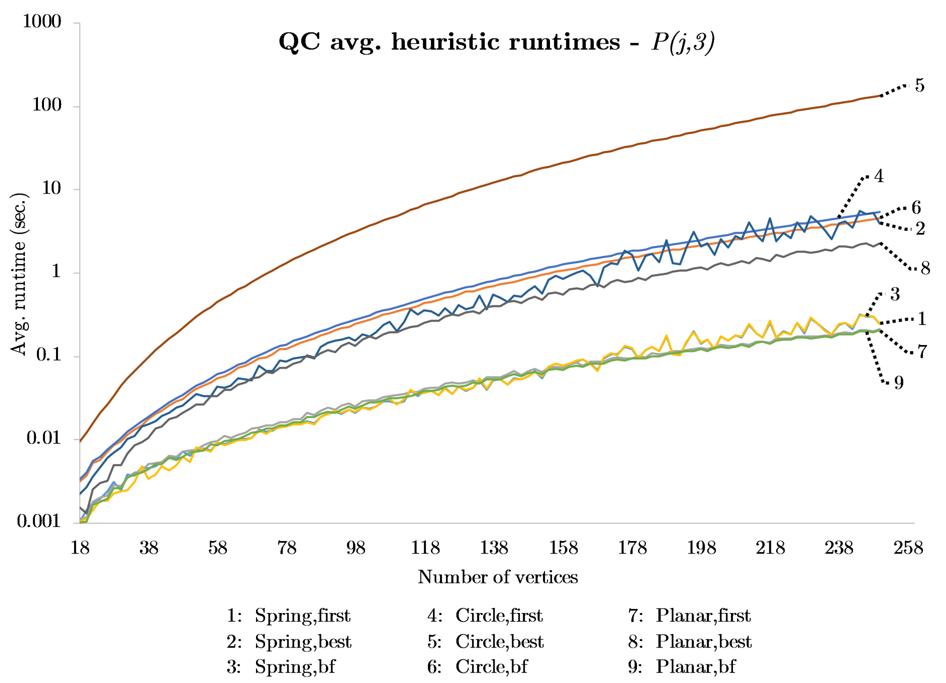


Figure 2.20: Average heuristic runtime of Quickcross per random permutation (sec.) compared to number of vertices for the graphs $P(j,3)$.

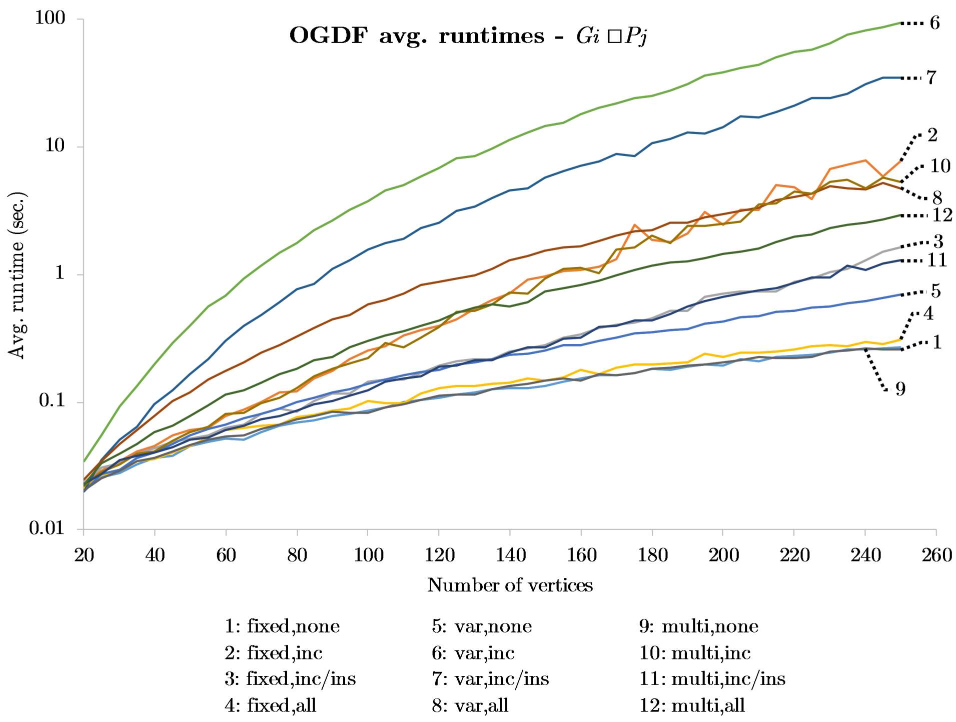


Figure 2.21: Average runtime of OGDF per random permutation (sec.) compared to number of vertices for the graphs $G_i \square P_j$.

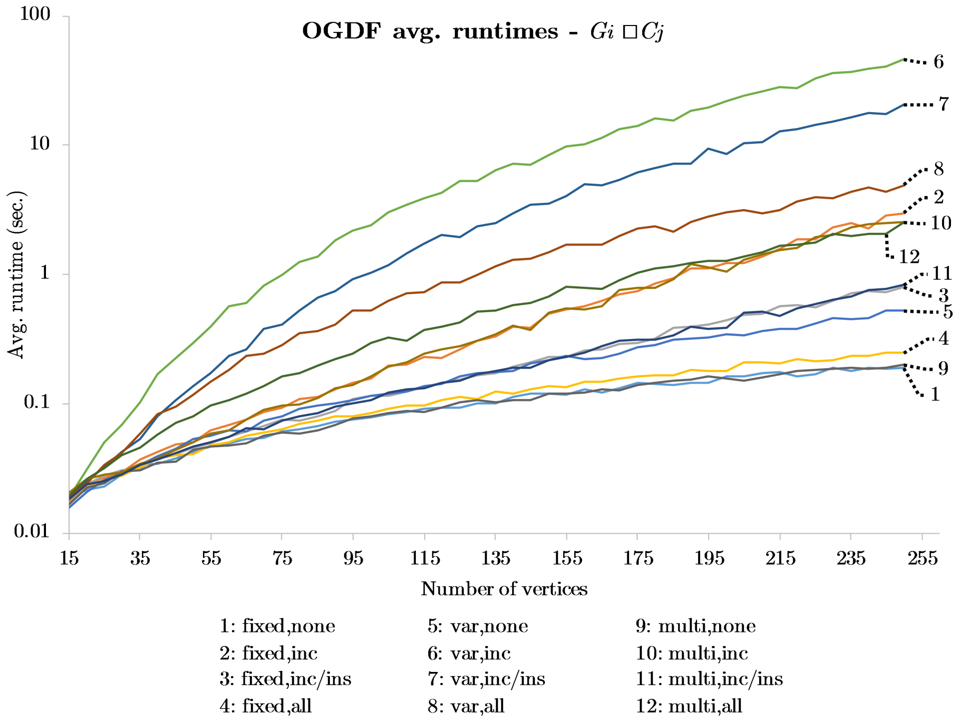


Figure 2.22: Average runtime of OGDF per random permutation (sec.) compared to number of vertices for the graphs $G_i \Box C_j$.

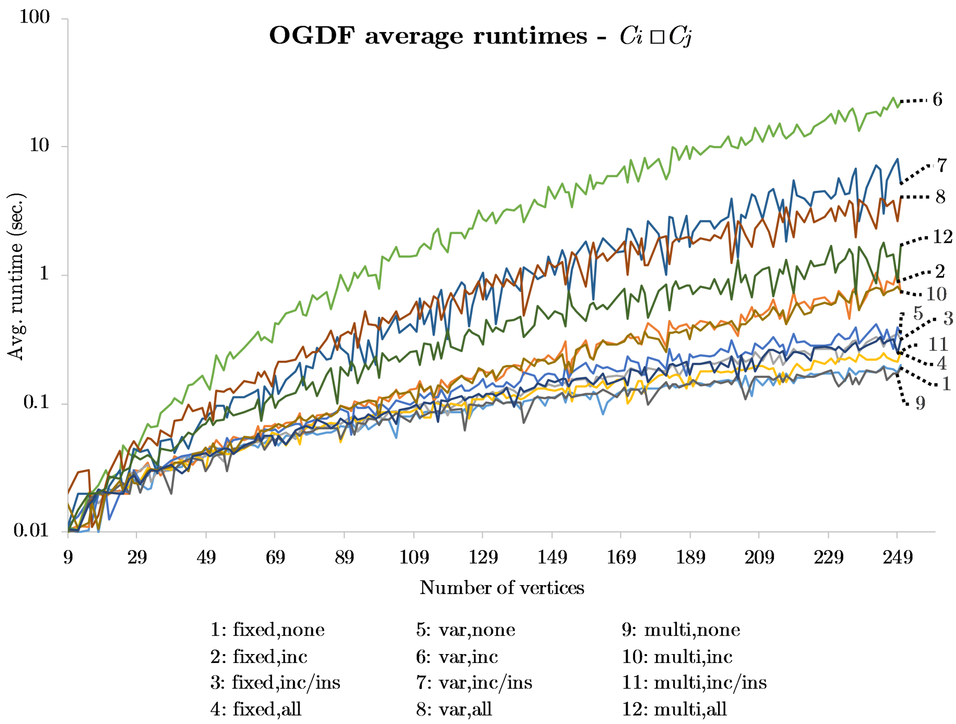


Figure 2.23: Average runtime of OGDF per random permutation (sec.) compared to number of vertices for the graphs $C_i \Box C_j$.

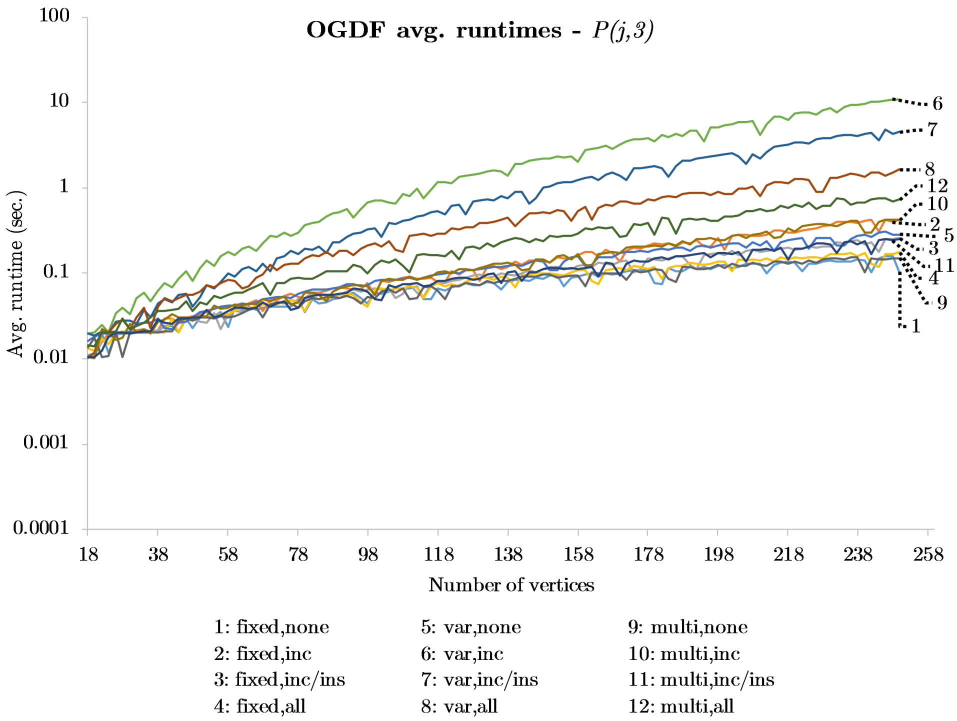


Figure 2.24: Average runtime of OGDF per random permutation (sec.) compared to number of vertices for the graphs $P(j,3)$.

which results in a significantly lower heuristic runtime.

KnownCR - Quickcross initial crossings (%)

Schemes	$G_i \square P_j$	$G_i \square C_j$	$C_i \square C_j$	$P(j,3)$
spring,first	382.80	1193.2	1134.6	338.23
spring,best	284.49	606.67	398.41	457.34
spring,bf	529.19	1354.2	989.87	280.59
circle,first	29783	24922	18300	27848
circle,best	29854	24704	18381	27512
circle,bf	29838	24788	18270	28092
planar,first	245.32	295.70	160.52	440.45
planar,best	241.04	300.70	160.30	447.63
planar,bf	244.68	297.30	160.78	441.48

Table 2.25: After only the initial embeddings of Quickcross for the KnownCR graphs. The values are the average percent relative deviation from the crossing number for the four families of graphs within the KnownCR set.

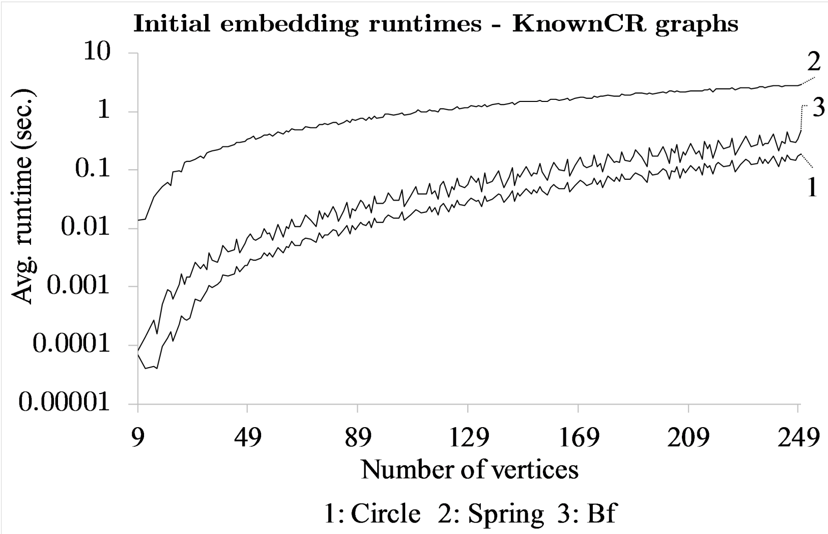


Figure 2.26: Average runtime to produce an initial embedding for Quickcross (sec.) compared to number of vertices for the KnownCR graphs.

2.5.3 Rome graphs

For the 140 graphs on 100 vertices in the Rome graph set, we repeat two experiments that have been previously performed in [65]. In the first experiment, for each of the possible combinations of heuristics and schemes, each graph is run with 100 random permutations. In the second experiment, the number of random permutations is increased to 500. In each case, we record the smallest number of crossings found for each graph, and report on the average of these values over the 140 graphs in Tables 2.27 and 2.28 respectively.

For Quickcross, we include average runtimes in Table 2.27, separated into time spent in the main part of the heuristic, and time spent in the initial embedding. The total average runtime is then the sum of these two values. Also, since the average runtime per permutation is almost identical after 100 permutations as it is after 500 permutations, we omit the runtimes from Table 2.28.

We note here that we encountered a bug with OGDF that prevented us from considering several of the Rome graphs with the *multi* scheme which we were unable to resolve. For this reason, we have omitted it from the experiment.

For Quickcross, we observe that after 100 random permutations, the *planar* embedding scheme outperforms both *circle* and *spring* in both solution quality and average runtime. Interestingly, the *best* minimisation scheme outperforms the other minimisation schemes under the same embedding scheme in each case. This result is different to the KnownCR graphs in which the *best* scheme was usually the worst performing scheme. Overall, the OGDF *var,inc* scheme obtained the best results with an average of 25.514 crossings, albeit with a runtime many times higher than for the other schemes in Quickcross. Then, with the sole exception of *var,inc*, every configuration of Quickcross compares favourably to the OGDF results. After 500 random permutations, in Table 2.28 the situation is slightly different. The Quickcross *circle,best* now outperforms the other Quickcross schemes, and again *var,inc* obtains the best result overall.

2.5.4 AT&T graphs

For the 311 graphs in the AT&T set, we report on a different experiment to the one previously performed in [31]. We treat this set in a similar manner as the Rome graphs. That is, each graph was run with 100 random permutations and in each case, we record the smallest number of crossings found, and then report on the average of these values over the 311 graphs. Then, we repeat the experiment and increase the number of random permuations to 500. These results are displayed in Tables 2.29 and 2.30 respectively.

After 100 random permutations, for OGDF, similarly to the experiments on the Rome graphs, *var,inc* performs very well on these graphs again with a

Rome - 100 random permutations

Method	Avg. final crossings	Avg. heuristic runtime (sec.)	Avg. initial crossings	Avg. embed runtime (sec.)
planar,best	25.757	0.0368	54.664	0.0061
planar,first	25.779	0.0099	54.664	0.0061
planar,bf	25.800	0.0110	54.664	0.0061
circle,best	25.829	0.2547	919.87	0.0029
spring,best	25.850	0.0469	80.971	0.2591
circle,bf	25.886	0.0425	919.87	0.0029
circle,first	25.900	0.0369	919.87	0.0029
spring,first	25.950	0.0105	80.971	0.2591
spring,bf	25.964	0.0118	80.971	0.2591
var,inc	25.514	0.6305		
var,all	26.021	0.2326		
fixed,inc	26.364	0.0648		
fixed,all	26.814	0.0473		
var,inc/ins	28.321	0.2648		
var,none	30.543	0.0603		
fixed,inc/ins	30.550	0.0459		
fixed,none	32.886	0.0432		
multi,none	-	-		
multi,inc	-	-		
multi,inc/ins	-	-		
multi,all	-	-		

Table 2.27: Averages of the minimum found crossings over 100 permutations for the Rome graphs run with the various schemes of Quickcross and OGDF. Average runtimes per random permutation are also displayed. The results are ordered by the average crossings and separated into Quickcross schemes and OGDF schemes.

Rome - 500 random permutations

Method	Avg. final crossings	Method	Avg. final crossings
circle,best	25.157	var,inc	25.093
planar,best	25.214	var,all	25.321
planar,first	25.229	fixed,inc	25.642
planar,bf	25.250	fixed,all	26.000
circle,first	25.300	var,inc/ins	27.121
spring,best	25.307	var,none	28.843
circle,bf	25.313	fixed,inc/ins	28.921
spring,bf	25.407	fixed,none	30.964
spring,first	25.457		

Table 2.28: Averages of the minimum found crossings over 500 permutations for the Rome graphs run with the various schemes of Quickcross and OGDF. The results are ordered by the average crossings and separated into Quickcross schemes and OGDF schemes.

higher runtime. The higher runtimes of OGDF's incremental schemes are now evident because of the much larger number of vertices for the graphs in the AT&T set. With the exception of *var,inc*, Quickcross obtains uniformly fewer crossings than OGDF. The *planar* scheme performs very well here, while the *circle* scheme performs the poorest. Again, similarly to the Rome set, *best* outperforms the other minimisation schemes under the same embedding scheme in each case. After 500 random permutations, in Table 2.30, the situation is very similar.

AT&T - 100 random permutations

Method	Avg. final crossings	Avg. total runtime (sec.)
planar,best	107.5981	0.4196
planar,bf	107.6881	0.0935
spring,best	107.701	1.133
planar,first	107.7042	0.0868
spring,bf	108.0064	0.4282
circle,best	108.0354	3.9598
circle,bf	108.0868	0.2999
spring,first	108.1736	0.4284
circle,first	108.2315	0.3342
var,inc	107.8167	14.3989
var,all	110.0707	2.432
multi,all	110.1608	0.9793
multi,inc	110.6367	1.6127
fixed,inc	111.1222	2.0537
fixed,all	113.1125	0.0928
var,inc/ins	121.9936	5.5283
multi,inc/ins	125.1608	0.371
fixed,inc/ins	128.1061	0.3877
var,none	134.6431	0.1157
multi,none	138.3312	0.0479
fixed,none	140.9775	0.0453

Table 2.29: Averages of the minimum found crossings over 100 permutations for the AT&T graphs run with the various schemes of Quickcross and OGDF. Average runtimes per random permutation are also displayed. The results are ordered by the average crossings and separated into Quickcross schemes and OGDF schemes.

AT&T - 500 random permutations

Method	Avg. final crossings	Method	Avg. final crossings
planar,best	106.75	var,inc	106.98
spring,best	106.83	var,all	108.26
planar,first	106.84	multi,all	108.46
planar,bf	106.85	multi,inc	109.06
spring,bf	107.03	fix,inc	109.36
circle,best	107.15	fix,all	110.97
spring,first	107.20	var,inc/ins	118.41
circle,first	107.20	multi,inc/ins	121.43
circle,bf	107.26	fix,inc/ins	123.64
		var,none	130.17
		multi,none	132.33
		fix,none	135.18

Table 2.30: Averages of the minimum found crossings over 500 permutations for the AT&T graphs run with the various schemes of Quickcross and OGDF.

2.5.5 ISCA graphs

The ISCA set is comprised of only 20 graphs, which means that it is possible to make comparisons for each individual graph. Each graph was run with 100 random permutations and in each case, we record the smallest number of crossings found. These results are displayed in Table 2.31 with the 20 graphs ordered by their number of vertices. Then, in Table 2.32, we display average runtimes, but only for those schemes which were successful in attaining the fewest crossings in several of the graphs.

Continuing the previous trend, for many of the graphs in this set, OGDF *var,inc* obtained the best result, again at the expense of significantly higher runtimes. On occasion, *var,inc* was beaten by either schemes from Quickcross or other OGDF schemes. As the number of vertices increases, it becomes apparent that the best results are obtained either from Quickcross, or OGDF *inc* or *all* and the other OGDF schemes perform significantly worse. Interestingly, there does not appear to be any one scheme in Quickcross which outperforms the others in general.

ISCA - 100 random permutations

Vertices	25	46	48	86	109	134	138	142	143	148	150	158	173	176	180	188	210	211	212	223
spring,bf	13	73	37	35	231	102	108	218	839	236	147	165	226	324	98	614	477	588	1202	860
spring,first	14	73	37	35	228	101	104	220	829	235	146	171	224	331	98	619	477	598	1219	848
spring,best	13	71	36	35	222	101	108	220	821	236	148	169	229	320	96	617	487	610	1218	860
circle,bf	13	69	36	35	227	108	103	214	837	232	147	165	230	324	95	606	495	607	1228	850
circle,first	13	75	38	36	231	105	109	225	825	236	148	173	235	328	95	615	508	605	1235	855
circle,best	13	73	38	35	230	103	109	224	840	242	154	167	230	319	100	627	477	598	1240	873
planar,bf	13	73	37	35	226	104	108	216	835	236	148	156	231	320	94	609	468	595	1200	853
planar,first	13	73	37	36	221	102	103	227	826	240	147	166	231	325	99	608	485	587	1232	859
planar,best	13	71	36	35	226	103	107	215	832	238	147	156	228	318	94	617	493	595	1235	869
fixed,none	14	92	46	46	304	144	155	318	1456	330	203	241	325	535	162	1003	788	891	2079	1572
fixed,inc	13	74	37	35	227	103	99	223	874	237	153	154	231	323	96	621	493	581	1264	908
fixed,inc/ins	13	87	43	43	271	128	135	273	1229	294	181	205	294	427	136	828	652	744	1607	1231
fixed,all	13	79	38	35	239	102	108	222	942	251	154	163	245	339	101	639	523	595	1284	953
var,none	14	94	44	44	293	137	141	290	1424	321	183	208	307	494	149	965	745	870	1954	1474
var,inc	13	70	37	35	225	99	99	214	842	231	142	154	224	315	93	600	474	570	1200	860
var,inc/ins	13	81	41	40	267	125	123	253	1137	285	172	180	268	397	125	757	594	712	1554	1158
var,all	13	74	39	35	225	102	102	222	905	241	149	159	218	332	96	629	495	583	1250	915
multi,none	13	94	46	47	303	138	138	290	1480	319	211	213	304	525	160	927	786	896	2007	1492
multi,inc	13	77	37	35	229	102	104	226	888	236	151	165	231	328	96	631	496	587	1261	897
multi,inc/ins	13	86	42	40	267	125	125	267	1189	289	177	175	281	409	129	791	610	746	1577	1213
multi,all	13	72	37	35	230	100	103	224	913	242	144	169	226	320	96	617	495	595	1243	902

Table 2.31: Minimum found crossings over 100 permutations for the ISCA graphs run with the various schemes of Quickcross and OGDF. The graphs are ordered by the number of vertices and the lowest crossings found for each graph are highlighted.

Runtime (sec.) for ISCA graphs

Vertices	25	46	48	86	109	134	138	142	143	148
spring,first	0.114	0.305	0.299	0.655	1.139	1.302	1.399	1.588	2.361	1.668
spring,best	0.116	0.390	0.339	0.929	3.208	3.258	3.203	5.167	18.19	5.762
circle,bf	0.011	0.069	0.050	0.280	1.030	1.008	1.237	1.752	4.267	1.883
planar,bf	0.007	0.032	0.017	0.065	0.310	0.218	0.243	0.418	1.571	0.490
planar,first	0.005	0.029	0.016	0.060	0.272	0.201	0.224	0.377	1.348	0.447
fixed,inc	0.023	0.156	0.062	0.136	2.572	0.711	0.736	5.501	112.3	4.550
var,inc	0.069	1.865	0.623	1.413	31.052	13.95	15.18	42.54	620.9	50.42
var,all	0.041	0.484	0.223	0.477	6.376	2.365	2.641	6.928	86.79	7.530

Vertices	150	158	173	176	180	188	210	211	212	223
spring,first	1.519	1.712	2.085	2.106	2.021	2.806	2.996	3.156	4.435	4.282
spring,best	4.280	5.562	7.881	8.902	6.718	22.39	21.49	24.39	60.31	44.90
circle,bf	1.686	1.996	2.650	3.399	2.792	6.455	5.523	4.527	9.503	8.566
planar,bf	0.343	0.428	0.619	0.722	0.445	1.777	1.375	1.361	3.640	2.541
planar,first	0.312	0.379	0.574	0.662	0.414	1.561	1.242	1.227	3.127	2.335
fixed,inc	1.155	2.790	2.733	17.62	1.514	146.7	46.88	91.04	362.1	247.3
var,inc	38.84	31.56	44.73	91.16	24.12	641.9	230.5	417.3	1461	1071
var,all	4.439	3.827	7.437	14.22	4.113	67.82	43.39	50.10	179.6	111.2

Table 2.32: Average total runtime per random permutation for the ISCA graphs for the schemes of Quickcross and OGDF that achieved the fewest crossings in several of the graphs.

2.5.6 Complete graphs

Recall that the crossing number of the complete graph K_n is conjectured (e.g. see Guy [68]) to be equal to

$$H(n) := \frac{1}{4}\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor\left\lfloor\frac{n-2}{2}\right\rfloor\left\lfloor\frac{n-3}{2}\right\rfloor.$$

Although this conjecture is widely believed to be correct, it has only been confirmed for $n \leq 12$ despite considerable effort to extend the results further [113]. We ran the graphs K_n for $20 \leq n \leq 50$. Each graph was run with 100 random permutations and the minimum found solution was compared to $H(n)$ by computing the percent relative deviation from $H(n)$. These results are displayed in Tables 2.33 for some selected values of n, and the runtimes are provided in Table 2.34.

For these graphs, we observe that when n was odd, every Quickcross scheme combination was able to obtain a drawing with $H(n)$ crossings. However, when n was even, each scheme reached a value which was usually very close but not equal to $H(n)$. The average runtime under the *best* scheme is significantly higher than the other minimisation schemes simply due to the vast amount of additional work required to consider every vertex each iteration.

Considering the runtimes for Quickcross, although *best* minimisation scheme became very slow, interestingly, this was not the case when the *planar* scheme is used. We conclude that for complete graphs, the *planar* scheme produces an initial embedding which is either optimal or near-optimal and hence very little additional work is required by Quickcross.

Conversely, for OGDF, these graphs quickly became intractable for almost all heuristic and scheme combinations, other than *fix,none*, *fix,all* and *multi,none*. For the schemes involving *none*, the results tended to be significantly worse than those of Quickcross. The *fix,all* scheme seemed to perform relatively well. We note here that the *fix,all* method is probably the closest analogue to Quickcross' *planar* initial embedding scheme, which might explain why these two were the best performing schemes for the respective platforms.

2.5.7 Complete bipartite graphs

Recall that the crossing number of the complete bipartite graph K_{n_1,n_2} is conjectured (e.g. see Zarankiewicz [147]) to be equal to

$$Z(n_1, n_2) := \left\lfloor \frac{n_1}{2} \right\rfloor \left\lfloor \frac{n_1 - 1}{2} \right\rfloor \left\lfloor \frac{n_2}{2} \right\rfloor \left\lfloor \frac{n_2 - 1}{2} \right\rfloor .$$

We ran the graphs K_{n_1,n_2} for $20 \leq n_1 \leq n_2 \leq 40$. Each graph was run with 100 random permutations and the minimum found solution was compared to $Z(n_1, n_2)$. For the sake of neatness, we only report on the cases where n_1 and n_2 are multiples of five. As can be seen in Table 2.35, Quickcross

Final crossings (%) for K_n

n	20	25	30	35	40	45	50
spring,first	0	0	0.0105	0	0.0185	0	0.0169
spring,best	0	0	0.0209	0	0.0246	0	*
spring,bf	0	0	0.0209	0	0.0246	0	0.0229
circle,first	0	0	0.0209	0	0.0185	0	0.0169
circle,best	0	0	0	0	0.0154	0	0.0145
circle,bf	0	0	0.0419	0	0.0400	0	0.0507
planar,first	0	0	0.0209	0	0.0062	0	0.0024
planar,best	0	0	0.0209	0	0.0062	0	0.0024
planar,bf	0	0	0.0314	0	0.0092	0	0.0036
fix,none	1.1111	1.2167	1.5175	1.4651	1.5327	1.2780	1.3502
fix,inc	0	0	*	*	*	*	*
fix,inc/ins	0.0617	0	*	*	*	*	*
fix,all	0.0617	0	0.0837	0	0.0615	0.0149	0.0700
var,none	1.0493	1.124	1.5489	1.1083	1.6189	1.4448	1.5410
var,inc	0	*	*	*	*	*	*
var,inc/ins	0	*	*	*	*	*	*
var,all	0.0617	0	0.0523	0	*	*	*
multi,none	1.1111	1.0330	1.4128	1.2867	1.6343	1.3361	1.3586
multi,inc	0	0	*	*	*	*	*
multi,inc/ins	1.2345	0	*	*	*	*	*
multi,all	0.0617	0	0.0837	0	0.0769	*	*

Table 2.33: Percent relative deviations from $H(n)$ after the conclusion of the heuristic, for the complete graphs K_n. A * entry indicates that the average runtime exceeded 3600 seconds (1 hour) per random permutation.

was successful in obtaining the conjectured optimum in all cases and for all scheme combinations, except $K_{30,30}$ under the *circle, best* combination. We suspect that these graphs are relatively easy for Quickcross to obtain a high-quality solution. However, although the conjectured optimum is easily reached for these graphs, the runtimes in Table 2.36 are comparable to those for the complete graphs, due to edge density. Again, the *best* minimisation scheme is significantly slower than the alternatives.

For OGDF, these graphs were intractable for the *inc* and *inc,ins* schemes. For the *none* and *all* schemes, OGDF is able to obtain drawings in comparable time to Quickcross, but those drawing almost never meet the conjectured value.

Runtime (sec.) for K_n

n	20	25	30	35	40	45	50
spring,first	0.4892	1.6260	7.3329	16.315	62.251	114.74	281.45
spring,best	2.2851	10.023	63.482	176.81	795.11	1551.0	*
spring,bf	0.5481	1.7314	7.8227	18.533	64.075	116.97	300.73
circle,first	0.4027	1.4541	6.8077	15.724	60.706	101.98	273.07
circle,best	2.1862	10.758	60.490	180.51	749.52	1215.2	3582.3
circle,bf	0.4636	1.7077	8.0522	21.293	69.723	131.98	346.20
planar,first	0.4093	0.6793	3.6526	6.5259	25.419	34.650	96.993
planar,best	0.5584	0.6924	8.3924	6.0631	76.533	35.386	266.51
planar,bf	0.4248	0.7250	3.3097	6.7406	22.067	35.586	86.218
fix,none	0.0781	0.2150	0.7258	1.7337	4.4118	8.4319	14.820
fix,inc	203.33	1592.0	*	*	*	*	*
fix,inc/ins	74.282	619.71	*	*	*	*	*
fix,all	0.8629	4.9050	17.277	59.869	172.24	476.42	617.88
var,none	1.6687	8.2514	39.804	131.24	284.97	597.59	1143.3
var,inc	1047.3	*	*	*	*	*	*
var,inc/ins	349.72	*	*	*	*	*	*
var,all	42.380	246.32	737.99	3068.9	*	*	*
multi,none	0.0788	0.2160	0.7288	1.7451	4.4177	8.3940	14.761
multi,inc	153.81	2411.3	*	*	*	*	*
multi,inc/ins	67.079	527.92	*	*	*	*	*
multi,all	21.269	89.570	302.82	975.74	2099.9	*	*

Table 2.34: Average total runtime (sec.) per random permutation for the complete graphs K_n. A * entry indicates that the average runtime exceeded 3600 seconds (1 hour) per random permutation.

2.5.8 Concluding observations

We now conclude this section with some general observations about the results of the various experiments.

With regards to Quickcross, although there is no one scheme that consistently outperforms the others, the *planar* initial embedding scheme and *first* minimisation scheme appears to offer the best overall performance in terms of both solution quality and runtime. The significantly increased runtime for the *best* minimisation scheme makes it impractical in general, but for extremely sparse graphs it performed well.

In general, Quickcross was able to produce solutions which were comparable or superior to all schemes of OGDF except for *var,inc*. However, *var,inc*

Final crossings (%) for K_{n_1,n_2}

n_1	20	20	20	20	20	25	25	25	25	30	30	30	35	35	40
n_2	20	25	30	35	40	25	30	35	40	30	35	40	35	40	40
spring,first	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
spring,best	0	0	0	0	0	0	0	0	0	0	0	0	0	*	*
spring,bf	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
circle,first	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
circle,best	0	0	0	0	0	0	0	0	0	0.0068	0	0	*	*	*
circle,bf	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
planar,first	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
planar,best	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
planar,bf	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
fix,none	2.4814	2.7777	2.8994	3.0449	2.7631	1.9917	2.6752	2.3692	2.3154	1.9546	2.2227	2.8634	1.6427	2.1070	1.8441
fix,inc	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
fix,inc/ins	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
fix,all	0.0987	0.0617	0.1058	0.0807	0.1403	0	0.0429	0.0048	0.0493	0.1043	0.0675	0.1102	0.0167	0.0491	0.0851
var,none	2.3209	2.8858	2.8677	3.1910	2.9385	1.7891	1.9775	2.7056	2.7887	2.1111	2.4501	2.5488	1.7348	2.5186	1.8261
var,inc	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
var,inc/ins	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
var,all	0.0493	0.0385	*	*	*	*	*	*	*	*	*	*	*	*	*
multi,none	2.3456	2.7391	2.9894	2.3952	2.5175	2.0158	2.4933	2.2202	2.6937	2.0793	2.4204	2.5827	1.8558	2.3693	1.9591
multi,inc	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
multi,inc/ins	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
multi,all	0.0864	0.0308	0.1216	0.0653	*	0	*	*	*	*	*	*	*	*	*

Table 2.35: Percent relative deviations from $Z(n_1, n_2)$ after the conclusion of the heuristic, for the complete bipartite graphs K_{n_1,n_2}.

Heuristic runtime (sec.) for K_{n_1,n_2}

n_1	20	20	20	20	20	25	25	25	25	30	30	30	35	35	40
n_2	20	25	30	35	40	25	30	35	40	30	35	40	35	40	40
spring,first	3.1301	5.3673	9.3235	15.715	22.948	12.591	19.816	33.160	49.168	45.149	67.594	92.526	141.27	187.85	393.21
spring,best	24.850	53.770	116.81	192.55	300.50	188.43	338.89	517.27	879.04	915.71	1238.7	1925.7	2548.7	*	*
spring,bf	4.1103	10.731	20.064	30.804	55.086	16.078	40.980	58.867	119.40	74.101	143.14	273.44	200.47	388.13	685.17
circle,first	2.7852	5.6527	9.7037	16.782	24.373	12.075	20.947	36.536	53.658	37.475	70.844	95.720	120.78	184.59	277.56
circle,best	46.978	68.300	127.07	216.52	335.43	291.24	439.63	617.86	892.52	1357.0	1668.3	1959.1	*	*	*
circle,bf	3.2240	6.1096	11.824	18.604	32.130	11.656	22.286	37.052	59.222	49.897	86.183	137.66	133.22	220.59	373.73
planar,first	2.6001	4.5518	9.4720	14.096	23.882	7.5417	18.898	23.348	48.137	40.114	62.963	108.55	78.906	184.33	320.16
planar,best	8.1185	13.159	35.332	40.687	91.321	9.1574	71.689	32.698	191.67	259.05	322.86	670.27	131.09	849.52	2260.4
planar,bf	2.5655	4.6000	9.2496	13.653	23.897	7.5333	19.686	23.965	48.871	41.858	66.725	113.64	84.707	201.86	335.96
fix,none	0.6887	1.2967	2.1732	3.6439	5.1076	2.5412	4.1690	7.0424	9.8109	8.0518	12.382	20.352	20.860	32.088	44.327
fix,inc	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
fix,inc/ins	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
fix,all	22.808	51.584	91.735	221.81	298.95	112.06	274.86	490.40	758.78	423.63	824.13	1436.9	1270.4	2656.7	3056.9
var,none	26.445	71.351	122.47	209.59	297.79	146.05	244.00	534.38	781.72	577.02	896.40	1385.9	1434.3	2063.4	2472.8
var,inc	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
var,inc/ins	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
var,all	551.20	1159.2	*	*	*	*	*	*	*	*	*	*	*	*	*
multi,none	0.6964	1.2856	2.3680	3.5732	5.1923	2.6229	4.6052	6.8399	10.204	7.6471	14.856	21.490	23.866	33.540	45.537
multi,inc	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
multi,inc/ins	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
multi,all	383.6705	699.1598	1818.1964	2442.055	*	1375.739	*	*	*	*	*	*	*	*	*

Table 2.36: Average heuristic runtime (sec.) per random permutation for the complete bipartite graphs K_{n_1,n_2}.

almost always produced superior results to any Quickcross scheme, as long as it was tractable. It appears that dense graphs provide a significant challenge for the more elaborate schemes within OGDF that makes them unsuitable. Conversely, it appears that performance of Quickcross scales relatively well with the edge density of the graph for all schemes except for *circle* and, to a lesser extent, *best*.

We note here that Quickcross answers the question posed by Chimani and Gutwenger in [31] about the performance of a heuristic based upon the star/vertex insertion problem, albeit only in the fixed embedding setting. From the experiments, it appears that for sparse graphs, Quickcross functions as an intermediate approach between the simplest implementation of the planarisation method, and the more sophisticated implementations like *var,inc*. That is, Quickcross obtains solutions which are significantly better than the standard planarisation method without a significant sacrifice in runtime, and similarly is significantly faster than the *var,inc* implementation without a significant sacrifice in solution quality. For dense graphs, Quickcross appears to be uniformly better than current implementations of the planarisation methods.

Nonetheless, the strong performance of the various *inc* schemes on sparse graphs motivated us to consider introducing this feature into Quickcross as well. In the following, final section of this chapter, we discuss an initial implementation of this idea. As will be seen, the results that it produces are comparable to those of OGDF's *var,inc* scheme. Designing a fully optimised version of this heuristic is a ripe topic for future research.

2.6 Incremental post-processing

We now investigate the computationally heavy post-processing strategy introduced for the planarisation method in [31] which is denoted *incremental* *post-processing*. Recall that in the planarisation method, one begins with a maximal planar subgraph, and the remaining edges of the graph are added to the drawing, one at a time. Then, the incremental post-processing strategy augments this approach in the following way. After each edge is added to the drawing, all existing edges of the drawing are deleted and reintroduced, one at a time, potentially resulting in fewer crossings. Obviously this significantly increases the amount of work to be done, but as seen in Section 2.5, the final result is often significantly better.

In Quickcross, by contrast, we do not introduce each edge one at a time to the drawing. Instead, we start with a complete drawing. Hence, a strategy such as the one above is not directly applicable to Quickcross. However, we note that for the *planar* embedding scheme, we introduce one vertex at a time. Hence, we can adapt the *planar* embedding scheme as follows. After each iteration of the *planar* embedding scheme we have an embedding Π corresponding to a drawing D_H of some connected vertex induced subgraph $H \subseteq G$. Then, recall that one of the options for Quickcross is to provide an initial drawing. As such, we can use this option to then run a new instance of Quickcross on the graph H using D_H as the initial drawing. Quickcross then attempts to minimise crossings and if it succeeds in finding a drawing of H with fewer crossings, we update Π accordingly and continue with the next iteration of the *planar* embedding scheme. We shall refer to this scheme as *Q-inc*.

We now repeat the experiments in Sections 2.5.2–2.5.5 with the new *Q-inc* strategy. In these experiments, we restrict ourselves to 100 random permutations. Note that during this process Quickcross is run independently on many subgraphs of G, and so we may select which minimisation scheme is used each time. Hence, in the upcoming experiments, we compare the three possible minimisation schemes (*first*,*best* and *bf*).

The runtime of this strategy is not analysed here because our current implementation is rudimentary, compared to the significant effort spent making the rest of Quickcross efficient. In its current stage of development, we simply make the comment that the runtime of the incremental post processing strategy is (as expected) significantly higher than any of the alternative strategies for Quickcross.

Table 2.37 contains the results for *Q-inc* on the KnownCR graphs. For the graph families $G_i \Box C_j$ and $C_i \Box C_j$, we observe a significant improvement compared to the results in Table 2.16. Notably, optimal drawings are obtained for all of the $C_i \Box C_j$ graphs. For the other two graph families, the results obtained by *Q-inc* are better than the results for the *planar* schemes given in Table 2.16. However, those two graph families were outperformed by the *spring* scheme in Table 2.16, and that continues to be the case for *Q-inc*. It appears that since *Q-inc* is based on the planar embedding scheme, there is no reason to assume that it will outperform the other schemes if the *planar* scheme did not already do so. However, we note that the *planar* embedding scheme was almost always the best performing scheme, and so *Q-inc* is still a promising approach.

Final crossings (%) for KnownCR graphs

Method	$G_i \Box P_j$	$G_i \Box C_j$	$C_i \Box C_j$	$P(j, 3)$
Q-inc,first	4.6688	0.8402	0	6.5585
Q-inc,best	4.6226	0.8127	0	6.6172
Q-inc,bf	4.6051	0.9010	0	6.6421

Table 2.37: Average percent relative deviations from the crossing numbers for the families within the KnownCR graphs for the *Q-inc* strategy.

Table 2.38 contains the results for *Q-inc* on the Rome graphs with 100 vertices. All three minimisation schemes produced superior results to any of those reported in Table 2.28. Notably, the results obtained by *Q-inc* here are superior even to those results obtained in Table 2.33 where 500 random permutations were considered.

Rome - 100 random permutations

Method	Avg. final crossings
Q-inc,first	25.000
Q-inc,bf	25.029
Q-inc,best	25.043

Table 2.38: The average number of crossings found over 100 permutations for the graphs on 100 vertices in the Rome graphs for the *Q-inc* strategy. The list is sorted by smallest average number of crossings.

Table 2.39 contains the results for *Q-inc* on the AT&T graphs. Surprisingly, for these graphs, all three minimisation schemes obtained the identical minimal number of crossings for each graph in the set. We expect that this result is a consequence of the *planar* embedding scheme upon which *Q-inc* is based, and that the various minimisation schemes have very little effect for these graphs. The results obtained by *Q-inc* are superior to those reported in Table 2.29, and were comparable to the best results in Table 2.30 where 500 random permutations were considered.

AT&T - 100 random permutations

Method	Avg. final crossings
Q-inc,first	106.849
Q-inc,bf	106.849
Q-inc,best	106.849

Table 2.39: The average number of crossings found over 100 permutations for the AT&T graphs for *Q-inc* strategy.

Table 2.40 contains the results for *Q-inc* on the ISCA graphs. We observe that, similarly to the AT&T graphs, all three minimisation schemes obtained the identical minimal number of crossings for each graph in the set. In Table 2.40, we compare the results of *Q-inc* to the most successful scheme on these graphs, *var,inc*, and also one of the best schemes from Quickcross, *planar,bf*. Notably, *Q-inc* obtains fewer crossings than all other tested schemes from Table 2.31 for 12 out of 20 graphs and, for some of the larger graphs in this set, the *Q-inc* solution is far superior to the next best solution. Additionally, *Q-inc* obtains superior results compared to *var,inc*, which was the best performing scheme, for 11 out 20 of the ISCA graphs.

ISCA graphs - 100 random permutations

Vertices	25	46	48	86	109	134	138	142	143	148
Q-inc,first	13	72	36	35	222	100	105	211	813	227
Q-inc,bf	13	72	36	35	222	100	105	211	813	227
Q-inc,best	13	72	36	35	222	100	105	211	813	227
var,inc	13	70	37	35	225	99	99	214	842	231
planar,bf	13	73	37	35	226	104	108	216	835	236

Vertices	150	158	173	176	180	188	210	211	212	223
Q-inc,first	145	156	216	301	94	598	461	572	1175	835
Q-inc,bf	145	156	216	301	94	598	461	572	1175	835
Q-inc,best	145	156	216	301	94	598	461	572	1175	835
var,inc	142	154	224	315	93	600	474	570	1200	860
planar,bf	148	156	231	320	94	609	468	595	1200	853

Table 2.40: Minimum found crossings over 100 permutations for the ISCA graphs run with the *Q-inc* strategy. The graphs are ordered by the number of vertices.

The results of the experiments for *Q-inc* are stark; there is significant improvement to be obtained by using this kind of post-processing scheme. Indeed, it appears that the results obtained by *Q-inc* are on par with *var,inc*, the best performing scheme from OGDF. However, much like *var,inc*, it comes at a significant computational cost. Investigating how best to implement this approach in an efficient manner is a clear task for future research. In addition, it is worthwhile investigating whether there is some analogue of incremental post-processing that may be applied to the other embedding schemes in Quickcross, for those graphs where the *planar* embedding scheme is not the most suitable.

Chapter 3

New exact results relating to crossing numbers

In Chapter 2, we introduced a new crossing minimisation heuristic. As discussed in that chapter, we have also developed a highly optimised implementation of this heuristic in C, and named this implementation Quickcross. The implementation includes pre-processing strategies, and can be applied to any undirected graph, irrespective of its connectivity or other properties. By modifying the various parameters and the random seed, we can obtain many different drawings of the same graph, and then select the drawing with the fewest crossings. If we consider sufficiently many random seeds, and the graph is not too complex, it is reasonable to expect that the number of crossings in this drawing is close, or even equal to, the crossing number of the graph.

Of course, Quickcross has no way of confirming whether or not it has found the crossing number of a graph. Nonetheless, the output from Quickcross does have some value. First, it provides a (hopefully reasonably tight) upper bound on the crossing number of the graph, which may be useful in some circumstances. We will describe some such situations below. Second, if Quickcross has been run many times, and regularly finds the same minimum number of crossings, we can be reasonably confident that this is equal to the crossing number. This then provides insight into what should be strived for in a subsequent proof. This approach is particularly enlightening when considering infinite families of graphs, for which the crossing numbers may obey a formula which is not obvious from the outset.

In this chapter, we will take advantage of the output of Quickcross. In particular, we will address the following problems.

In Section 3.1, we consider two open conjectures of Pegg Jr. and Exoo from 2009, about the size of the smallest cubic graphs which have crossing number at least k. Specifically, the conjectures are about the cases $k = 9, 10, 11$. We will demonstrate that Quickcross is ideal for resolving these conjectures. This is because doing so involves considering a very large set of graphs (around 430 million), but in this case we do not require their exact crossing numbers. Rather, we simply need to confirm that the crossing numbers are below a certain value. The upper bounds provided by Quickcross are suitable for such a purpose.

Then, in Section 3.2, we study a family of graphs arising from the Cartesian product of a Sunlet graph and a star. In particular, we seek to determine the crossing numbers of this family of graphs. It is not clear from the outset what this formula should be, let alone how to prove it. However, the output from Quickcross, leads to a conjecture about the formula for the crossing numbers. We then prove that the conjecture is true for some cases where the size of the star is fixed.

Finally, in Section 3.3, we consider the crossing numbers of families of graphs which result from a graph product of given small graphs with certain arbitrarily large graph families. This has been a pursuit of researchers for several decades, since the crossing number of the Cartesian product of each connected 4-vertex graph with an arbitrarily large cycle was determined in [19]. Researchers have since extended this by replacing cycles with stars and paths, and considering larger fixed graphs. They have also considered other kinds of graph products, most notably join products involving cycles, paths and discrete graphs. For graph products of these arbitrarily large graphs with graphs on four vertices, these have now all been determined. However, for the graphs with more than four vertices, many gaps remain, and researchers have been slowly filling these gaps, often on a case-by-case basis with ad-hoc proofs.

For a specific case, the proof typically involves establishing both an upper and lower bound, and then showing that these coincide. The latter is typically much more difficult than the former. However, in some cases, a lower bound from a related family can be utilised. This has been done in some isolated cases. However, here we use Quickcross to perform a broad empirical study of all cases involving graphs up to six vertices, in order to identify in a systematic way all such situations where this is possible. As a result, we are able to prove the crossing numbers of 29 new families of graph products. This is a notable achievement, since over the last several decades, and hundreds of individual papers, only 207 such results have been previously determined.

3.1 Minimal cubic graphs with crossing number at least k

Suppose that, for a chosen integer k, we are interested in graphs with crossing number at least k. A question that one might ask is what is the smallest number of vertices such a graph can have. If we restrict our consideration to cubic graphs (for which the crossing number problem is still NP-hard [79]), some results are known. Define a_k to be the order of the smallest cubic graph with crossing number at least k. In Table 3.1 we list the values a_k for $k = 1, \dots, 8$, along with an example of one such cubic graph of that order, and the number of minimal examples; these are taken from Pegg Jr and Exoo [114]. It should be noted that in [114], it was claimed that there are five minimal examples of cubic graphs with crossing number at least 8, however it has been subsequently determined during private communication between Pegg Jr and Eric Weisstein that two of them (labelled in [114] as CNG 8D and CNG 8E) were erroneously listed, and the correct number is three.

k	a_k	**Example**	**# minimal examples**
0	4	K_4	1
1	6	$K_{3,3}$	1
2	10	Petersen graph	2
3	14	Heawood graph	8
4	16	Möbius-Kantor graph	2
5	18	Pappus graph	2
6	20	Desargues graph	3
7	22	Unnamed graphs, see Figure 3.2	4
8	24	McGee graph	3

Table 3.1: The minimum number of vertices of a cubic graph which possesses crossing number at least k, a named example of each, and the number of minimal examples.

It is also worth noting at this point that the crossing numbers provided for many of the graphs listed in Table 3.1, although widely accepted as accurate and listed as such in numerous sources, have never been formally established in literature. We remedy that here, by using the excellent exact crossing minimisation solver of Chimani and Wiedera [38] to confirm that all of the minimal examples (not just the named ones) listed in Pegg Jr and Exoo [114] have their crossing numbers correctly listed, other than CNG 8D and CNG 8E as previously noted.

Results on minimal cubic graphs with crossing number larger than 8 have, to date, only been conjectured. It has been widely accepted that the Coxeter graph [41] on 28 vertices has crossing number 11, and the Levi graph [101] (also known as the Tutte-Coxeter graph) on 30 vertices has crossing number

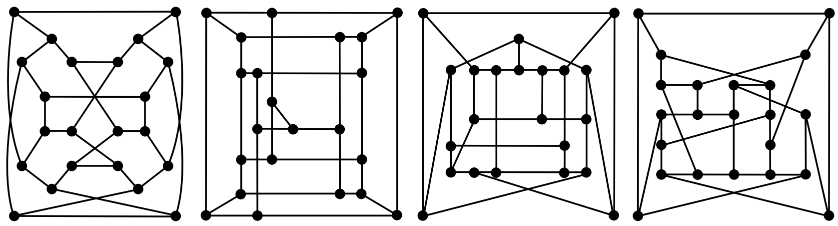


Figure 3.2: The four unnamed graphs on 22 vertices which constitute examples of the smallest cubic graphs possessing crossing number 7. These graphs and their drawings are taken from [114].

13, although again these results have not been formally established in literature. We again remedy this oversight here by reporting that the exact solver [38] confirms that these crossing numbers are accurate. Then, an open question posed by Pegg Jr and Exoo [114] is whether any cubic graphs of order 26 have crossing number 11. More precisely, they conjectured the following.

Conjecture 3.1 (Pegg Jr and Exoo, 2009 [114]). *With a_k as defined above,*

(i) $a_9 = a_{10} = 26.$

(ii) $a_{11} = 28.$

(iii) $a_{13} = 30.$

In what follows, we answer questions (*i*) and (*ii*). Specifically, we show that (i) is false, and that (ii) is true. One corollary of these results is that the Coxeter graph is a minimal example of a cubic graph with crossing number 11. At this stage, we have not answered question (*iii*), but at the end of this section we briefly discuss how it might be attacked.

We now outline the approach that we take to resolve question (*i*) and (*ii*) of Conjecture 3.1. First, denote by $f(n)$ the largest crossing number of any cubic graph on n vertices. Values of $f(n)$ for small n can be deduced from the known values of a_k and these are displayed in Table 3.3. An important point is that, in general, it is not necessarily the case that $f(n)$ is equal to the largest k such that $a_k \leq n$. In other words, given a value of a_k, k is not necessarily the largest crossing number among all cubic graphs on a_k vertices. However, the following observation can be made.

$\boldsymbol{n}$	4	6	8	10	12	14	16	18	20	22
$\boldsymbol{f(n)}$	0	1	1	2	2	3	4	5	6	7

Table 3.3: The largest crossing number $f(n)$, for any cubic graph on n vertices.

Lemma 3.2. *For any even $n \geq 4$, $f(n+2) \geq f(n)$.*

Proof. Let G be a cubic graph on n vertices which attains $f(n)$ crossings. Subdivide any two distinct edges of G and join the newly created vertices with an edge, call the resulting graph G'. Then, $f(n) \leq cr(G') \leq f(n+2)$. □

Next, we consider the case of non-simple connected 3-regular graphs, and show that there always exists a simple connected 3-regular graph of the same order with crossing number at least as large. This result will be important in the upcoming Lemma 3.4 and Proposition 3.5.

Lemma 3.3. *Consider any non-simple connected 3-regular graph G on $n \geq 4$ vertices. Then, $cr(G) \leq f(n)$.*

Proof. Suppose that we possess an optimal drawing D of G. Since G is non-simple, it must contain some number of multiedges and loops. It is clear from the optimality of D that there are no crossings on the loops in D. Hence, any loops can be deleted without altering the crossing number.

Next, we consider multiedges. Since G is 3-regular, there can be at most three multiedges between any pair of vertices u and v. If there are three, then u and v constitute the whole graph and $n = 2$. Hence, we only need to consider the case where there are two multiedges between u and v.

Suppose that one of the multiedges is crossed more times than the other in D. Then, it is possible to redraw that multiedge so that it is arbitrarily close to the other multiedge, which reduces the number of crossings. This contradicts the optimality of D. Hence, both multiedges are crossed the same number of times. This, in turn, implies that D can be modified so that these two multiedges lie arbitrarily close to each other without altering the number of crossings. Note that this also implies that both multiedges are crossed by the identical set of edges. We refer to this modified, but still optimal, drawing as D'.

Finally, consider any edge (or multiedge) e which crosses both multiedges between u and v in D'. Since G is 3-regular, it is clear that another, single, edge is incident to u, and likewise there is another edge incident to v. Hence, e can be redirected to cross one of these single edges instead, and the number of crossings reduces. This contradicts the optimality of D', and hence we conclude that there are no crossings on the multiedges. Hence, one of the multiedges may be removed without altering the crossing number.

By applying the above to every instance of multiedges and loops, we obtain a simple subcubic graph G' on n vertices with the same crossing number as G. It is clear that there exists a cubic graph on at most n vertices with which G' is a subdivision of, and hence $cr(G) = cr(G') \leq f(n)$. □

The next Lemma is crucial in reducing the computational work required to determine the new values of a_k. For the sake of dealing with the boundary cases in the below argument, we extend $f(n)$ so that $f(2) = 0$ and $f(0) = 0$.

Lemma 3.4. *Any cubic graph G on n vertices with girth 3 has crossing number $cr(G) \leq f(n - 2r)$, where r is the number of triangles in G.*

Proof. Consider any cubic graph G with girth 3, we shall construct a drawing of G with at most $f(n - 2r)$ crossings. If G is not K_4, then each triangle of G either contains zero edges which are involved in a second triangle, or a single edge involved in a second triangle, in which case the two triangles form a diamond. Each diamond of G connects two vertices u and v which are not part of the diamond. For each diamond of G, delete it and connect u and v with a new edge. For each other triangle of G, contract it into a single vertex. The result is a new 3-regular (not necessarily simple) graph G' on $n-2r$ vertices. Hence, $cr(G') \leq f(n-2r)$. Note from Lemma 3.3 that this holds whether or not G' is simple. Now, consider an optimal drawing D' of G'. For those vertices in D' which were the result of a contracted triangle, replace them with an arbitrarily small triangle such that there are no crossings on the triangle. Similarly for those edges in D' which were the result of a replaced diamond, replace them with an arbitrarily small diamond connected to the corresponding end vertices such that there are no crossings on the diamond. These operations do not increase the number of crossings and so the result is a drawing of G with at most $f(n-2r)$ crossings, hence $cr(G) \leq f(n-2r)$. □

We are now in the position to present the values of a_9, a_{10} and a_{11}. We first discuss a_9. We observed that one of the examples given in [114] of a cubic graph on 26 vertices, which is drawn there with 10 crossings, actually has crossing number 9. The graph is dubbed the 'McGee graph plus an edge', meaning that a pair of edges of the McGee graph were subdivided and the newly created vertices joined by an edge. A reproduction of the drawing given in [114] is displayed in Figure 3.4. The crossing number for this graph has been verified to be 9 using the exact crossing minimisation solver [38]. Figure 3.5 shows the McGee graph plus an edge drawn with 9 crossings, which we now know to be an optimal drawing.

Proposition 3.5. *$a_9 = 26$.*

Proof. By the above discussion, there exists at least one cubic graph on 26 vertices with crossing number 9. Hence $a_9 \leq 26$.

To show the reverse inequality, we need to establish that $f(24) < 9$. To achieve this, we performed a series of computations using Quickcross. Firstly,

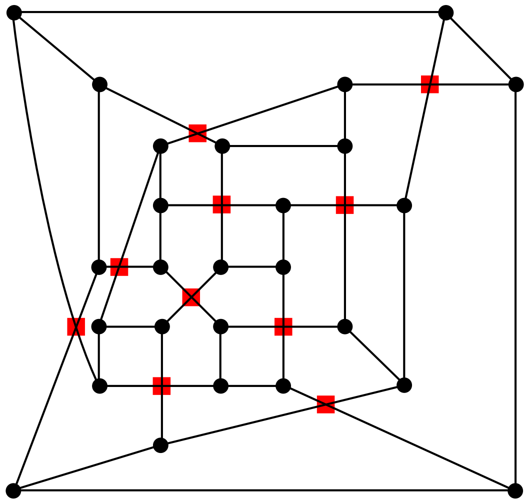


Figure 3.4: The cubic graph, from [114], dubbed 'McGee graph plus an edge' drawn with 10 crossings. Crossings are highlighted in red.

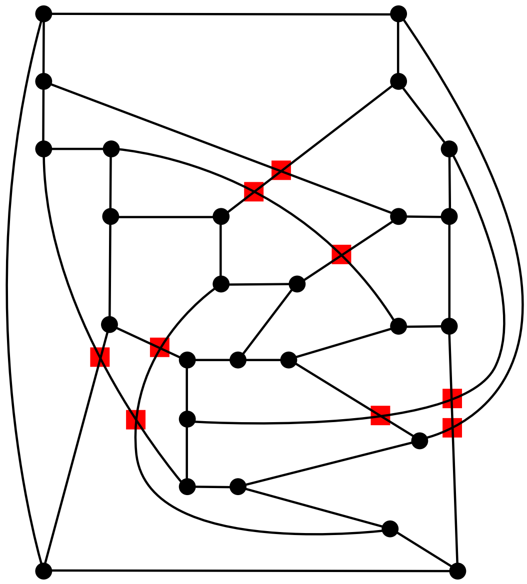


Figure 3.5: The cubic graph dubbed 'McGee graph plus an edge' drawn with 9 crossings, which is an optimal drawing. Crossings are highlighted in red.

for any cubic graph G on 24 vertices with girth 3, G has at least one triangle and so by Lemma 3.4 and Table 3.3, $cr(G) \leq f(22) = 7$. For the other cubic graphs on 24 vertices of girth 4 or greater (of which there are 23,780,814), Quickcross was successful in obtaining drawings with strictly fewer than 9 crossings. Therefore we can conclude that $f(24) \leq 8$, and Lemma 3.2 now implies that for any $n \leq 24$, $f(n) \leq 8$. Therefore, we obtain $a_9 \geq 26$. □

As a side note, the lower bound in Proposition 3.5, along with the known value of a_8, provides the following.

Corollary 3.6. $f(24) = 8$.

The proof of Proposition 3.5 relied on Quickcross producing valid drawings for each graph. Although we are very confident in Quickcross functioning correctly, we nonetheless decided to verify that each drawing found was indeed valid. To do this, we planarised each of the drawings produced by Quickcross. If the drawings are valid, then the planarisation necessarily produces a planar graph. In each case, we then used the planarity checking algorithm of Hopcroft and Tarjan [83] to confirm that the planarisation was indeed planar. For all graphs tested, this was the case.

Eliminating the girth 3 graphs from consideration in Proposition 3.5 reduced the number graphs we needed to consider from 117,940,535 down to 23,780,814 (roughly an 80% reduction). Eliminating the girth 3 graphs becomes imperative when considering the next largest case (26 vertex graphs) below.

We now discuss the values of a_{10} and a_{11}. Similarly to Proposition 3.5, we eliminate the girth 3 graphs when considering cubic graphs on 26 vertices. This again reduces the number of graphs needing considered by roughly 80% from 2,094,480,864 down to 432,757,568. Then, for each of these graphs, we used Quickcross to find drawings with optimal or near-optimal number of crossings. Unlike the 24 vertex case, this is no longer tractable on a normal computer. For cubic graphs on 26 vertices, Quickcross can process several graphs per second, including time taken to read and write data to the disk. We partitioned the graphs into sets of 50,000, resulting in roughly 8,000 individual jobs, each of which took up to 3 days to run, and distributed the jobs over 400 cores on a High Performance Computer. Eventually we were able to obtain drawings with 9 or fewer crossings for all of the 26 vertex cubic graphs of girth 4 or larger. Similarly to the previous experiment, we again verified the results using the planarity checking algorithm of Hopcroft and Tarjan [83].

To provide an upper bound for a_{10}, we confirmed that there exists a cubic graph on 28 vertices which has crossing number 10. This unnamed graph is one of the 21 cubic graphs on 28 vertices with girth 7, which is the largest girth of cubic graphs of this order. An optimal drawing of this graph with 10 crossings is displayed in Figure 3.6. The crossing number was again confirmed by the exact crossing minimisation solver [38].

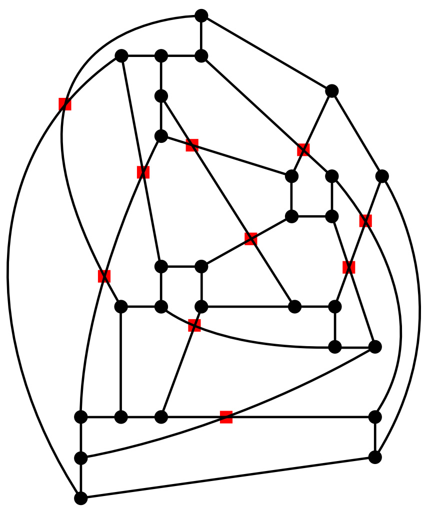


Figure 3.6: A cubic graph on 28 vertices, which has been confirmed to have crossing number 10, drawn optimally. The crossings are highlighted in red.

Proposition 3.7. $a_{10} = 28$.

Proof. From the above discussion, it is clear from the runs of Quickcross that $f(26) \leq 9$. Hence, $a_{10} \geq 28$. Then, having confirmed that the graph in Figure 3.6 has crossing number 10, we conclude that $a_{10} = 28$. $\square$

The lower bound in Proposition 3.7, along with the value of a_9 from 3.5 also provides the following.

Corollary 3.8. $f(26) = 9$.

Propositions 3.5 and 3.7 allow us to now answer question (i) of Conjecture 3.1 in the negative. In particular, we have shown that a_9 is equal to 26 as stated in Conjecture 3.1, but a_{10} is equal to 28, rather than 26. In addition, the results above allow us to immediately answer question (ii) of Conjecture 3.1 as well, in the affirmative.

Proposition 3.9. $a_{11} = 28$.

Proof. As discussed previously, the Coxeter graph on 28 vertices has crossing number 11. Hence $a_{11} \leq 28$.

To show the reverse inequality, from Corollary 3.8, we have shown that for any $n \leq 26$, $f(n) \leq 9$ and therefore $a_{11} \geq 28$. $\square$

A zip file is available upon request, which contains a list of edge crossings for each cubic graph on 26 vertices with girth 4 or more. The edge crossings correspond to a valid drawing with 9 or fewer crossings.

Some known additivity properties of the crossing number could have aided in the computations of this section. Specifically, results shown in [22] and [100] imply that the crossing number of any cyclically k-connected cubic graph, where $k \leq 3$ is the sum of the crossing numbers of its augmented components, when minimal cyclic edge cutsets have been deleted. It so happens that, in this case, the augmented components are homeomorphic to smaller cubic graphs. After handling some technicalities, it can be shown that the following relationship holds.

Lemma 3.10. *For any k and any $n_1, n_2, ..., n_k$ such that $n_1 + n_2 + ... + n_k \leq n + 2(k-1)$,*

$$f(n) \geq \sum_{i=1}^{k} f(n_i).$$

This means that we could have restricted our computations with Quickcross to cyclically-k-connected cubic graphs, where $k \geq 4$ (there is also one other easy case to consider). In particular, eliminating the 1-connected cubic graphs would have been advantageous. However, identifying cyclically-k-connected cubic graphs is non-trivial and we were satisfied with handing more graphs to Quickcross instead of complicating the calculations.

Lastly, we make a remark about question (*iii*) of Conjecture 3.1. The number of cubic graphs on 28 vertices is 40,497,138,011 and, of these, 8,542,471,494 have girth 4 or larger [25]. This a substantial increase from 26 vertices and there would be a large jump in computation time if our above methods were applied to (*iii*). However, we suspect that other innovations, similar to Lemma 3.4, could be used to make an answer to (*iii*) more tractable.

3.2 Cartesian product of a Sunlet graph and a Star graph

Crossing numbers have been determined for some infinite families of graphs. In many such cases, the family is created by taking the Cartesian product of members of two smaller graph families. To the best of our knowledge, the first publication along these lines was by Harary, Kainen and Schwenk [75] in 1973, who conjectured that the crossing number of $C_m \square C_n$, that is, the Cartesian product of two arbitrarily large cycles, would be $n(m-2)$ for $n \geq m \geq 3$. To date, this conjecture remains unproven, although a number of partial results have been determined. In a long line of research spanning several decades, results have been determined for small values where $m \leq 7$ [6, 10, 11, 19, 43, 75, 96, 118, 119, 121]. Also, in 2004, Glebsky and Salazar [62] provided a breakthrough by showing that for any m, the conjecture holds for all but a finite number of the n.

Theorem 3.11 (Glebsky and Salazar, 2004 [62]). *For any $m \geq 3$ and any $n \geq m(m+1)$, consider the cycle graphs C_n and C_m, then*

$$cr(C_n \Box C_m) = n(m-2).$$

Other infinite graph families, for which the crossing numbers of their Cartesian products have been studied, include paths and stars [85, 90, 21], complete graphs and cycles [141], cycles and stars [85, 90], wheels and trees [95], and cycles with the 2-power of paths [93]. Many of these results are described in greater detail in Section 3.3.

In this section, we expand this growing literature by considering the Cartesian product of a Sunlet graph and a star. The Sunlet graph on $2n$ vertices, denoted $\mathcal{S}_n$ for $n \geq 3$, is constructed by attaching n pendant edges to the n-cycle C_n; see Figure 3.7 for an example of $\mathcal{S}_6$. In order to avoid confusion with the notation for the Sunlet graph, we note that the star S_m is equivalent to the complete bipartite graph $K_{1,m}$, and so we use the latter notation for stars in this section. We will show that $cr(\mathcal{S}_n \Box K_{1,m}) \leq n\frac{m(m-1)}{2}$ for $n \geq 3$ and $m \geq 1$. We will also prove that the crossing number meets this bound precisely for $m \in \{1, 2, 3\}$, and conjecture that it does so for all $m \in \mathbb{Z}+$.

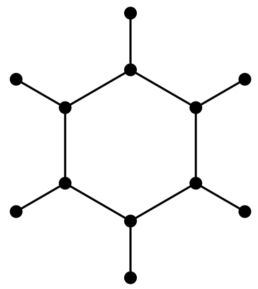


Figure 3.7: The Sunlet graph, $\mathcal{S}_6$.

3.2.1 Upper Bound

We begin by providing an upper bound for $cr(\mathcal{S}_n \Box K_{1,m})$. When we first considered this family, we constructed a number of small cases, and submitted them to Quickcross, with the intention of using its results to predict their crossing numbers. The results from Quickcross are displayed in Table 3.8, and it appears from those results that the crossing number of $\mathcal{S}_n \Box K_{1,m}$ is likely to be $n\frac{m(m-1)}{2}$.

	n							
m	3	4	5	6	7	8	9	10
2	3	4	5	6	7	8	9	10
3	9	12	15	18	21	24	27	30
4	18	24	30	36	42	48	54	60
5	30	40	50	60	70	80	90	100
6	45	60	75	90	105	120	135	150
7	63	84	105	126	147	168	189	210
8	84	112	140	168	196	224	252	280
9	108	144	180	216	252	288	324	360
10	135	180	225	270	315	360	405	450

Table 3.8: Best found solutions obtained by Quickcross for the graphs $\mathcal{S}_n \Box K_{1,m}$.

The first step in confirming that this is, indeed, the correct formula for the crossing numbers is to establish it as an upper bound. In order to do so, we need to provide a drawing procedure that meets this number of crossings.

In what follows, let the vertex labels of $K_{1,m}$ be v_0 for the vertex of degree

m and $v_1, v_2, \ldots, v_m$ for the vertices of degree 1. Let the vertex labels of $\mathcal{S}_n$ be $u_0, u_1, u_2, \ldots, u_{n-1}$ for the vertices on the cycle and let u_i' denote the pendant vertex attached to u_i.

Theorem 3.12. *The crossing number of $\mathcal{S}_n \Box K_{1,m}$ is no larger than $n\frac{m(m-1)}{2}$ for $n \geq 3$, $m \geq 1$.*

Proof. It is easy to check that $\mathcal{S}_n \Box K_{1,1}$ is planar; for instance, a planar drawing of $\mathcal{S}_6 \Box K_{1,1}$ is illustrated in Figure 3.9, which can obviously be extended for any n. It then suffices to give a procedure for drawing the graph $\mathcal{S}_n \Box K_{1,m}$, $m \geq 2$, so that the number of crossings meets the proposed upper bound.

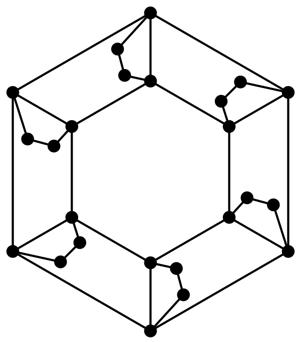


Figure 3.9: Planar drawing of $\mathcal{S}_6 \Box K_{1,1}$.

First, note that $\mathcal{S}_n \Box K_{1,m}$ contains $C_n \Box K_{1,m}$ as a subgraph. Begin by drawing the subgraph $C_n \Box K_{1,m}$ in the manner illustrated in Figure 3.10(a). For a given $i = 0, 1, \ldots, n-1$, the thick edges represent $((v_0, u_i), (v_j, u_i))$ for $j = 0, 1, \ldots, m$. The dashed edges represent $((v_j, u_i), (v_j, u_{i+1}))$ and $((v_j, u_i), (v_j, u_{i-1}))$ for $j = 0, 1, \ldots, m$. Then, it is easy to see that the dashed edges can be joined to the corresponding sections for $i+1$ and $i-1$, and so on, to complete a drawing of $K_{1,m} \Box C_n$ without introducing any additional crossings. Hence, the number of crossings in this drawing of the subgraph $C_n \Box K_{1,m}$ is:

$$n\left(\sum_{k=1}^{\lfloor \frac{m}{2} \rfloor - 1} k + \sum_{k=1}^{\lceil \frac{m}{2} \rceil - 1} k\right) = n \left\lfloor \frac{m}{2} \right\rfloor \left\lfloor \frac{m-1}{2} \right\rfloor. \tag{3.1}$$

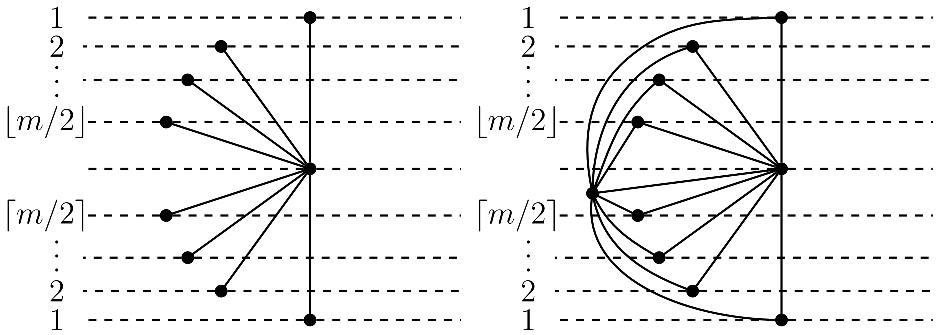


Figure 3.10: In (a), the construction of a drawing of the subgraph $K_{1,m}\square C_n$. In (b), the extension which will be subdivided to produce a drawing of $K_{1,m}\square \mathcal{S}_n$

Next, we extend this drawing to a drawing of $\mathcal{S}_n\square K_{1,m}$ in the following way. For each $i = 0, 1, \ldots, n-1$, place a vertex in the region between the centre horizontal (dashed) edge $((v_0, u_i), (v_0, u_{i+1}))$ and the first thick edge on the side which possesses $\lceil m/2 \rceil$ vertices, and join this new vertex to each of the vertices (v_j, u_i) for $j = 0, 1, \ldots, m$ as in Figure 3.10(b). Then, the number of crossings in this graph is equal to:

$$\begin{aligned} n\left(\left\lfloor \frac{m}{2} \right\rfloor \left\lfloor \frac{m-1}{2} \right\rfloor + \sum_{k=1}^{\lfloor \frac{m}{2} \rfloor} k + \sum_{k=1}^{\lceil \frac{m}{2} \rceil - 1} k \right) & \\ = n \left\lfloor \frac{m}{2} \right\rfloor \left(\left\lfloor \frac{m-1}{2} \right\rfloor + \left\lceil \frac{m}{2} \right\rceil \right) & \\ = n\frac{m(m-1)}{2}. & \end{aligned} \tag{3.2}$$

Finally, if every new edge is subdivided, except for the ones emanating from (v_0, u_i) for $i = 0, 1, \ldots, n-1$, the resulting graph is isomorphic to $\mathcal{S}_n\square K_{1,m}$. Since subdividing edges does not alter the number of crossings, we conclude that it is possible to draw $\mathcal{S}_n\square K_{1,m}$ with $n\frac{m(m-1)}{2}$ crossings. $\square$

3.2.2 Exact results

We now consider $\mathcal{S}_n\square K_{1,m}$ for some small values of m, and show that the crossing number coincides precisely with the upper bound from Section 3.2.1.

Denote that upper bound by $U(n,m) := n^{\frac{m(m-1)}{2}}$. As noted previously, $\mathcal{S}_n \square K_{1,1}$ is planar; see Figure 3.9. This agrees with $U(n,1) = 0$. Next, we will consider the cases when $m = 2$ and $m = 3$.

In what follows, we will utilise some properties of subgraphs of $\mathcal{S}_n \square K_{1,m}$, which we denote by H_i for each $i = 0, 1, 2, \ldots, n-1$. In particular, H_i is defined as the subgraph induced by the union of the following, disjoint, sets of edges:

$$\begin{aligned}
a_i &:= \{((v_j,u_i),(v_j,u_{i+1})) \mid j = 0,1,\ldots,m\} \\
b_i &:= \{((v_j,u_i),(v_j,u_i')) \mid j = 0,1,\ldots,m\} \\
b_i' &:= \{((v_0,u_i'),(v_j,u_i')) \mid j = 1,\ldots,m\} \\
c_i &:= \{((v_j,u_i),(v_j,u_{i-1})) \mid j = 0,1,\ldots,m\} \\
t_i &:= \{((v_0,u_i),(v_j,u_i)) \mid j = 1,\ldots,m\} \\
t_{i+1} &:= \{((v_0,u_{i+1}),(v_j,u_{i+1})) \mid j = 1,\ldots,m\} \\
t_{i-1} &:= \{((v_0,u_{i-1}),(v_j,u_{i-1})) \mid j = 1,\ldots,m\}
\end{aligned}$$

A detailed illustration of H_i, for the case $m = 3$, is displayed in Figure 3.11. For each $i = 0, 1, 2, \ldots, n-1$, there is a corresponding H_i in $\mathcal{S}_n \square K_{1,m}$ and H_i and H_j contain common edges when $j = i+1$ or $j = i-1$. The union of all H_i is precisely $\mathcal{S}_n \square K_{1,m}$.

We now consider the case when $m = 2$. Note that $U(n,2) = n$. In what follows, we use the following notation: consider a drawing D of a graph which contains two edge sets a and b. Then $cr_D(a)$ is equal to the number of crossings on the edges of a in D. Similarly, $cr_D(a,b)$ is equal to the number of crossings in D between edge-pairs, such that one edge is contained in a and the other is contained in b.

Lemma 3.13. *The crossing number of $\mathcal{S}_n \square K_{1,2}$ is equal to n.*

Proof. From Theorem 3.12, we know that $cr(\mathcal{S}_n \square K_{1,2}) \leq n$. Hence, the task now is to show that the reverse inequality holds. Let H_i' be the subgraph H_i without the edges t_i. An illustration of H_i' is displayed in Figure 3.12.

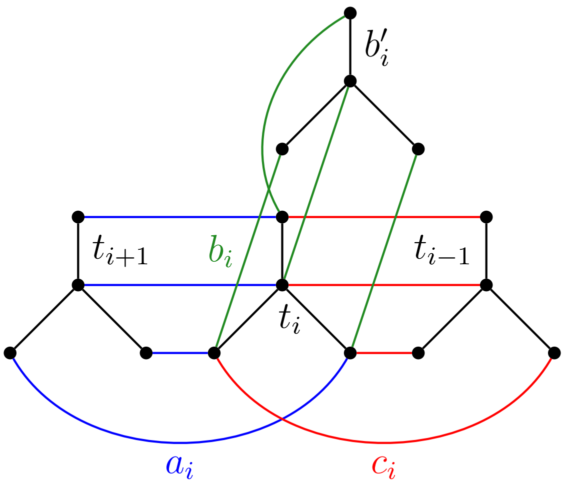


Figure 3.11: The subgraph H_i of $\mathcal{S}_n \Box K_{1,3}$. The labels for each set of edges lie next to one edge belonging to that set. In this drawing, the solid lines correspond to the sets t_{i-1}, t_i, t_{i+1} and b_i'.

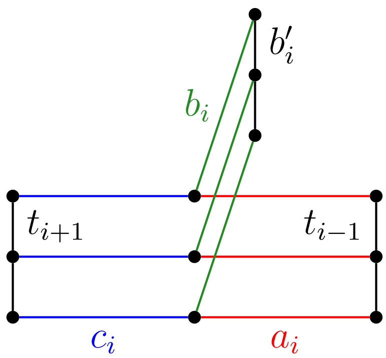


Figure 3.12: The subgraph H_i' of $\mathcal{S}_n \Box K_{1,2}$. The labels for each set of edges lie next to one edge belonging to that set.

It is clear that H_i' is homeomorphic to $K_{3,3}$, and hence there exists at least one crossing in the subdrawing D' of H_i'. Furthermore, at least one crossing in D' involves two edges which come from the edge sets $(a_i \cup t_{i+1}), (b_i \cup b_i')$ and $(c_i \cup t_{i-i})$, but do not both come from the same edge set. That is,

$$\begin{aligned} cr_{D'}\big((a_i \cup t_{i+1}), (b_i \cup b_i')\big) + cr_{D'}\big((a_i \cup t_{i+1}), (c_i \cup t_{i-1})\big) \\ + cr_{D'}\big((b_i \cup b_i'), (c_i \cup t_{i+1})\big) \geq 1. \end{aligned} \tag{3.3}$$

Hence, it is clear that there is at least one crossing in each H_i' which does not occur in any other H_j' for $i \neq j$, which leads immediately to the result. □

Next, we consider the case when $m = 3$. Note that $U(n, 3) = 3n$. In order to handle this case, we first need to prove two intermediate results, Lemmas 3.14–3.15.

Lemma 3.14. *For $m = 3$, consider the following four edge sets: $(a_i \cup t_{i+1})$, $(b_i \cup b_i')$, $(c_i \cup t_{i-1})$ and t_i. Then, in any good drawing of the subgraph H_i, there are at least 3 crossings for which the two edges involved in the crossing are not in the same set.*

Proof. H_i is homeomorphic to $K_{1,3,3}$, and Asano [13] proved that $\mathrm{cr}(K_{1,3,3}) = 3$. Any drawing of H_i corresponds to some drawing of $K_{1,3,3}$. Any drawing of $K_{1,3,3}$ has at least three crossings between pairs of edges which are not incident. These crossings correspond precisely to crossings in the drawing of H_i which satisfy the Lemma. □

Lemma 3.15. *For $n \geq 3$, let D be a drawing of $\mathcal{S}_n \square K_{1,3}$. If, for each $i = 0, 1, 2, \ldots, n-1$, the edges $t_i \cup b_i \cup b_i'$ are crossed two or fewer times in D, then D has at least $3n$ crossings.*

Proof. Let F_i denote the edge set $t_i \cup b_i \cup b_i'$. Note that F_i is a subgraph of H_i. Then, from Lemma 3.14, we have

$$\begin{aligned} cr_D\big(a_i \cup t_{i+1}, F_i\big) + cr_D\big(c_i \cup t_{i-1}, F_i\big) + cr_D\big((a_i \cup t_{i+1}), (c_i \cup t_{i-1})\big) \\ + cr_D\big(F_i, F_i\big) \geq 3. \end{aligned} \tag{3.4}$$

Assume that $cr_D(F_i) \leq 2$ for all $i = 0, 1, 2, \ldots, n-1$. It will be shown that if $cr_D\big(t_{i+1}, F_i\big) \neq 0$, or if $cr_D\big(t_{i-1}, F_i\big) \neq 0$, then a contradiction arises.

Suppose that $cr_D\big(t_{i+1}, F_i\big) = 1$. Note that the edges of b_{i+1} link to all of the endpoints of t_{i+1}. Since the subgraph induced by F_i is 2-connected, it is clear that it is impossible to draw $(b_{i+1} \cup b_{i+1}')$ without creating an additional crossing on the edges of F_i. Since the subgraph induced by $F_i \cup c_i \cup t_{i-1}$ is isomorphic to $P_2 \square S_3$, where P_2 denotes the path graph on 3 vertices, and $cr(P_2 \square S_3) = 1$ [85], the following also holds

$$cr_D\big(c_i \cup t_{i-1}, F_i\big) + cr_D\big(F_i, F_i\big) \geq 1.$$

This would imply that F_i is crossed at least three times, but by assumption, $cr_D(F_i) \leq 2$. Hence, $cr_D\big(t_{i+1}, F_i\big) \neq 1$. An analogous argument can be made for t_{i-1} which, similarly, implies that $cr_D\big(t_{i-1}, F_i\big) \neq 1$ as well.

Suppose then that $cr_D\big(t_{i+1}, F_i\big) = 2$. Then, since $cr_D(F_i) \leq 2$, it must be the case that $cr_D(F_i, F_i) = 0$, and hence without loss of generality, the subdrawing of the subgraph induced by F_i must be equivalent to the drawing displayed in Figure 3.13.

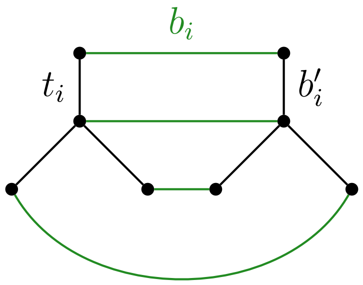


Figure 3.13: The drawing of the subgraph induced by F_i, if F_i is not crossed by itself.

Now consider the rest of the subgraph H_i, which includes edge sets $(a_i \cup t_{i+1})$ and $(c_i \cup t_{i-1})$. Note that the edges c_i link to all of the endpoints of t_i, and these do not lie on a common face of D, so $(c_i \cup t_{i-1})$ cannot be drawn without crossing F_i at least once. This would imply that F_i is crossed at least three times, but by assumption, $cr_D(F_i) \leq 2$. Hence, $cr_D\big(t_{i+1}, F_i\big) \neq 2$. An analogous argument can be made for t_{i-1} which, similarly, implies that $cr_D\big(t_{i-1}, F_i\big) \neq 2$ as well.

Then, since $cr_D(F_i) \leq 2$, the only remaining possibility is $cr_D\big(t_{i+1}, F_i\big) = cr_D\big(t_{i-1}, F_i\big) = 0$, and so (3.4) simplifies to

$$\begin{aligned} cr_D\big(a_i, F_i\big) + cr_D\big(c_i, F_i\big) + cr_D\big((a_i \cup t_{i+1}), (c_i \cup t_{i-1})\big) \\ + cr_D\big(F_i, F_i\big) \geq 3. \end{aligned} \tag{3.5}$$

It can be easily seen that any crossing counted by the left hand side of (3.5) is not counted for any other $j \neq i$. Hence summing (3.5) over $i = 0, 1, 2, \ldots, n-1$ provides the result. □

Finally, we are ready to propose the theorem for $m = 3$.

Theorem 3.16. *For $n \geq 3$, the crossing number of $\mathcal{S}_n \square K_{1,3}$ is equal to $3n$.*

Proof. We will prove the result by induction. The base case where $n = 3$, corresponding to a graph on 24 vertices, was proved computationally, utilising the exact crossing minimisation methods of Chimani and Wiedera [38]. The proof comes from a solution to an appropriately constructed integer linear program and shows that $cr(\mathcal{S}_3 \square K_{1,3}) = 9$. The proof file is available upon request.

Now, suppose that $cr(\mathcal{S}_n \square K_{1,3}) = 3n$ for each $n = 3, \ldots, k-1$, but that for $n = k$ there exists a drawing with strictly fewer than $3k$ crossings. Let D denote such a drawing. By Lemma 3.15, there must be at least one i such that the edges of F_i are crossed at least three times in D. Hence, the edges F_i could be deleted and the number of crossings remaining would be strictly less than $3(k-1)$. However, once F_i is deleted, the resulting graph is homeomorphic to $\mathcal{S}_{k-1} \square K_{1,3}$, which by the inductive assumption has crossing number equal to $3(k-1)$. This is a contradiction, and hence any drawing for $n = k$ must have at least $3k$ crossings. This, combined with Theorem 3.12 implies that $cr(\mathcal{S}_k \square K_{1,3}) = 3k$, and inductively we obtain the result. □

We conclude this section by conjecturing that the upper bound described in Theorem 3.12 coincides precisely with the crossing number in all cases.

Conjecture 3.17. *For $n \geq 3$, $m \geq 1$,*

$$cr(\mathcal{S}_n \Box K_{1,m}) = n\frac{m(m-1)}{2}$$

Conjecture 3.17 is supported by Table 3.8, which shows that Quickcross was able to find a drawing meeting this conjecture in all tested cases and was unable to find any counter-examples.

3.3 Crossing numbers of graph products involving one small graph

In this section, we significantly extend the known results of crossing numbers of graph families resulting from graph products. The first results in this vein were developed by Beineke and Ringeisen in [19]. There are six non-isomorphic connected graphs on four vertices. For each of these, Beineke and Ringeisen considered their Cartesian product with arbitrarily large cycles, and determined the crossing numbers of each of the six families. This research was later extended by Jendrol and Ščerbová [85] and as Klešč [92] who, instead of arbitrarily large cycles, considered arbitrarily large paths and stars. Subsequent researchers then attempted to consider graphs with more than four vertices, as well as other kinds of graph products, most notably the join product.

In the years since, the crossing numbers for many of the cases where the small graph has five or six vertices have been developed, often on a case-by-case basis, with ad-hoc proofs which exploit the structures of the specific graphs in question. However, there is still quite a number of open cases remaining to be determined. A recent comprehensive survey of graph families with known crossing numbers [39] contains a section dedicated to collecting all results of this kind that have been established to date. This includes well over a hundred papers employing all manner of techniques.

In general, the approach to establishing the crossing numbers for one such family is to provide an upper bound (typically via a drawing procedure), and then establish a lower bound, and show that these coincide. Establishing the lower bound is often quite technical, and is usually considered the difficult part of the proof. However, in some cases, it is possible to use the known lower bound from a related graph family to provide the lower bound for an undecided case. Of course, this only determines the crossing number if the lower bound so provided is tight, which occurs only when the crossing numbers for the two graph families coincide. This has been applied in some cases (e.g. see [94]), however not in a systematic way. Here, for the first time, we attempt to do so.

We note in advance that the proofs contained in this section are, for the most part, simple. However, they resolve a significant proportion of the cases which have remained open for decades, despite enormous effort worldwide and over a hundred published research papers in this area. The reason we are in the advantageous position to make such simple breakthroughs is twofold. First, the recent survey [39] has brought together, for the first time, the complete list of known results, so that they may all be leveraged simultaneously. Prior to the existence of this survey, it was common for the same graph families to be considered by multiple researchers, unaware that they had already been decided, or that related results existed that might be used. Second, since Quickcross is very reliable for small sparse graphs, we can efficiently predict, with a high degree of accuracy, when two graph families share the same crossing numbers. This enables us to determine, in a systematic fashion, precisely which open cases may be decided using existing results.

We begin with some observations about the crossing numbers of related graph families. Firstly, if H is a subgraph of G, then clearly $cr(H) \leq cr(G)$, which immediately gives the following relations for the Cartesian and join products:

Lemma 3.18. *If H is a subgraph of G, then for any graph F, $cr(H\Box F) \leq cr(G\Box F)$.*

Lemma 3.19. *If H is a subgraph of G, then for any graph F, $cr(H+F) \leq cr(G+F)$.*

Suppose that we have H which is a subgraph of G, and F_i which is the i-th member of some infinite family of graphs. If we already know the value of $cr(H\Box F_i)$ for all i, then by Lemma 3.18 that value constitutes a lower bound for $cr(G\Box F_i)$ as well. Then, if we can determine an upper bound for $cr(G\Box F_i)$ for each i that coincides with this lower bound, it follows immediately that $cr(G\Box F_i) = cr(H\Box F_i)$. However, this is obviously only applicable if $cr(G\Box F_i)$ does indeed coincide with the lower bound.

As discussed previously, this is where Quickcross is useful. We can use it to predict the crossing numbers for the graph families resulting from $H\Box F_i$ and $G\Box F_i$, and see if they are equal. In some cases, they may not be equal for the smallest members of the infinite families, but become equal after a certain size. When this is observed, it indicates that an upper bound can be found which coincides with the lower bound. Then, all that remains is to produce an appropriate drawing procedure to establish the upper bound. From Lemma 3.19 it is clear that the above discussion is also applicable to join products.

In what follows, we will consider graphs on five and six vertices, and so we now introduce some notation to help identify these graphs. Figure 3.14 displays all 21 non-isomorphic connected graphs on five vertices in the order given by Klešč [91]. Figure 3.15 displays all 156 non-isomorphic graphs on six vertices, in the order originally given in Frank Harary's classic textbook,

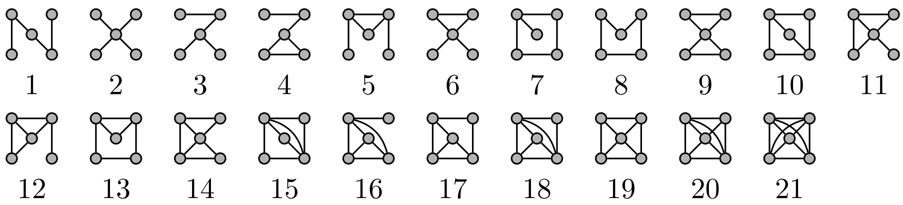


Figure 3.14: All 21 connected graphs on five vertices.

Graph Theory [74]. These graphs are ordered by their number of edges. Note that we include the 44 disconnected graphs on six vertices, so that the indices may remain consistent for potential future research purposes. The notation we use to identify these graphs is G_i^5 or G_i^6 to denote a graph on five or six vertices, where i is the index provided in Figures 3.14 and 3.15 respectively. Note that the Cartesian product of a disconnected graph G with another graph F results in a graph which is the union of the Cartesian products of each connected component of G with F. Hence, for Cartesian products, it is only necessary to consider connected graphs. Conversely, the join product always results in a connected graph. Previously, it has been rare for researchers to consider join products involving a small disconnected graph, nonetheless we include the disconnected graphs in Figure 3.15 for potential future research purposes.

We will begin by considering cases involving Cartesian products, for which all of our new results involve six vertex graphs. Then, in Section 3.3.2, we will consider cases involving join products, which will include both five vertex and six vertex graphs.

In the following Theorems, there are many examples of drawing procedures for graphs. Determining the resulting number of crossings in these drawings is usually an easy exercise and so we omit the proofs and instead provide one detailed example from Theorem 3.23 (one of the more complicated instances) in Appendix A.

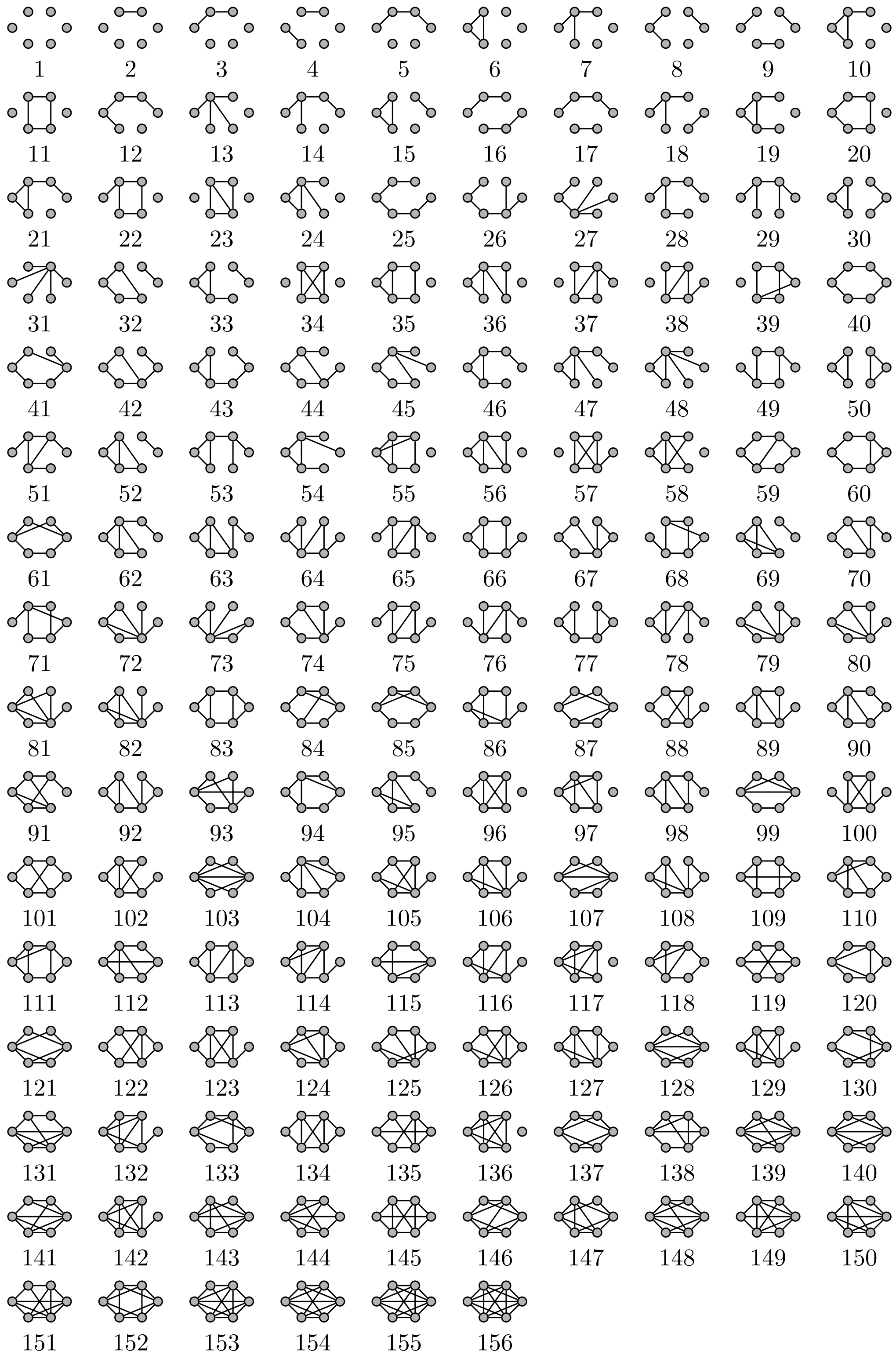


Figure 3.15: All 156 graphs on six vertices and their indices, ordered by number of edges.

3.3.1 Cartesian products involving one small graph

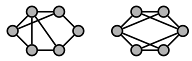
Six vertex graphs with paths

According to the recent survey of graph families with known crossing numbers [39], these cases have been decided for 57 of the six vertex graphs, and we reproduce the table of known results from [39] here as Table 3.16.

Table 3.16: Known crossing numbers of Cartesian products of graphs on six vertices with paths. All results are for $n \geq 1$.

i	G_i^6	$cr(G_i^6 \square P_n)$	i	G_i^6	$cr(G_i^6 \square P_n)$	i	G_i^6	$cr(G_i^6 \square P_n)$
25		0	60		$n-1$	90		$3n-3$
26		$n-1$	61		$2n$	93		$4n$
27		$2n-2$	64		$2n-2$	94		$2n-2$
28		$n-1$	65		$3n-3$	103		$6n-2$
29		$2n-2$	66		$2n-2$	104		$4n-4$
31		$4n-4$	67		$3n-3$	109		$4n$
40		0	70		$3n-3$	111		$3n-1$
41		$n-1$	72		$4n-4$	113		$4n-4$
42		$2n-4$	73		$4n-4$	118		$4n-2$
43		$n-1$	74		$2n-2$	119		$7n-1$
44		$2n-2$	75		$2n$	120		$3n-3$
45		$2n-2$	77		$2n-2$	121		$4n$
46		$n-1$	79		$4n-4$	125		$5n-3$
47		$2n-2$	80		$4n-4$	130		$4n$
48		$4n-4$	83		$2n-2$	146		$5n-1$
51		$3n-3$	84		$3n-1$	152		$6n$
53		$2n-2$	85		$2n$	154		$9n-1$
54		$2n-2$	86		$3n-1$	155		$12n$
59		$2n-2$	89		$3n-3$	156		$15n+3$

We will now determine new results, which can be added to Table 3.16, for the following two graphs.



G^6_{110} G^6_{137}

Theorem 3.20. *Consider the path graph P_n, then for $n \geq 1$,*

1. $cr(G^6_{110} \square P_n) = 3n - 1$.
2. $cr(G^6_{137} \square P_n) = 4n$.

Proof of 1. It can be seen that $G^6_{110} \supset G^6_{84}$, and so by Lemma 3.18, $cr(G^6_{110} \square P_n) \geq cr(G^6_{84} \square P_n)$. Also, from [140], we have $cr(G^6_{84} \square P_n) = 3n-1$. Figure 3.17 provides a drawing procedure for $G^6_{110} \square P_n$ with $3n-1$ crossings and therefore $cr(G^6_{110} \square P_n) = 3n - 1$.

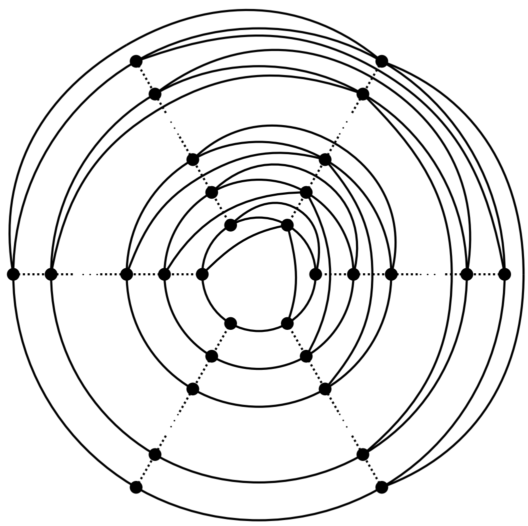


Figure 3.17: $G^6_{110} \square P_n$ drawn with $3n-1$ crossings (the -1 comes from the outer ring). The solid lines are the edges of copies of G^6_{110} and the dashed lines are the edges of copies of P_n.

Proof of 2. It can be seen that $G^6_{137} \supset G^6_{121}$ and so by Lemma 3.18, $cr(G^6_{137} \square P_n) \geq cr(G^6_{121} \square P_n)$. Also, from [94], we have $cr(G^6_{121} \square P_n) = 4n$. Figure 3.18 provides a drawing procedure for $G^6_{137} \square P_n$ with $4n$ crossings and therefore $cr(G^6_{137} \square P_n) = 4n$.

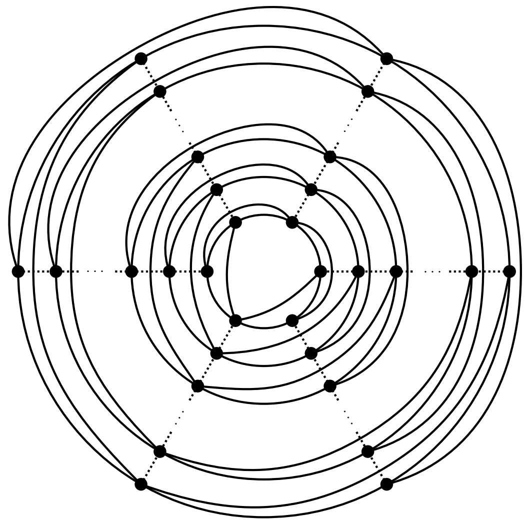


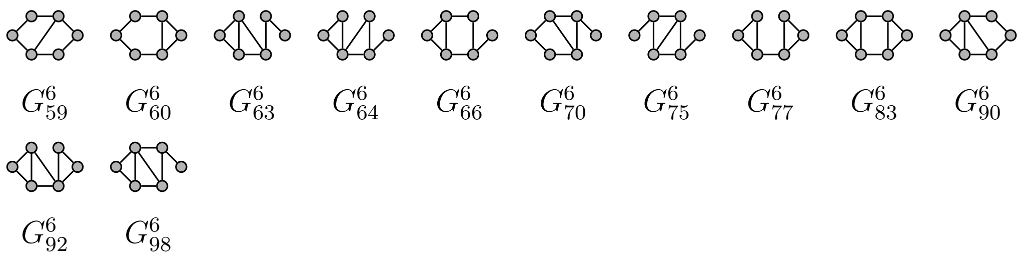
Figure 3.18: $G^6_{137} \Box P_n$ drawn with $4n$ crossings. The solid lines are the edges of copies of G^6_{137} and the dashed lines are the edges of copies of P_n.

$\square$

Six vertex graphs with cycles

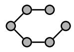
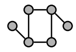
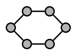
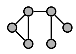
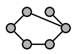
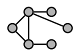
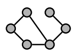
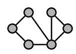
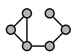
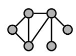
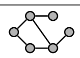
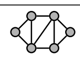
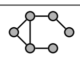
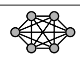
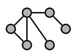
To date, results are only known for graphs possessing low edge density, with the only exception being the result with K_6. In total, according to [39], the crossing number of $G^6_i \Box C_n$ has been decided for 15 graphs to date. These results are reproduced in Table 3.19.

We will now determine new results, which can be added to Table 3.19, for the following twelve graphs.



Theorem 3.21. *Consider the cycle graph C_n, then for $n \geq 6$, $cr(G^6_{59} \Box C_n) = cr(G^6_{60} \Box C_n) = cr(G^6_{83} \Box C_n) = cr(G^6_{90} \Box C_n) = 4n$.*

Table 3.19: Crossing numbers of Cartesian products of graphs on six vertices with cycles.

i	G_i^6	$cr(G_i^6 \Box C_n)$	i	G_i^6	$cr(G_i^6 \Box C_n)$
25		0	49		$2n \quad (n \geq 4), \quad 4 \quad (n = 3)$
40		$4n \quad (n \geq 6)$, $6 \quad (n = 3)$ $12 \quad (n = 4)$ $18 \quad (n = 5)$	53		$2n \quad (n \geq 6)$, $4 \quad (n = 3)$ $6 \quad (n = 4)$ $9 \quad (n = 5)$
41		$3n \quad (n \geq 5)$, $5 \quad (n = 3)$ $10 \quad (n = 4)$	54		$2n \quad (n \geq 6)$, $4 \quad (n = 3)$ $6 \quad (n = 4)$ $9 \quad (n = 5)$
42		$2n \quad (n \geq 4), \quad 4 \quad (n = 3)$	67		$3n \quad (n \geq 4), \quad 7 \quad (n = 3)$
43		$n \quad (n \geq 3)$	78		$3n \quad (n \geq 6)$, $7 \quad (n = 3)$ $10 \quad (n = 4)$ $14 \quad (n = 5)$
44		$2n \quad (n \geq 4), \quad 4 \quad (n = 3)$	113		$4n \quad (n \geq 3)$
46		$n \quad (n \geq 3)$	156		$18n \quad (n \geq 3)$
47		$2n \quad (n \geq 6)$, $4 \quad (n = 3)$ $6 \quad (n = 4)$ $9 \quad (n = 5)$			

Proof. Consider the graphs G_{113}^6 and G_{40}^6. It is shown in [93] and [118] respectively that the crossing number of $G_{113}^6 \Box C_n$ for $n \geq 3$ and $G_{40}^6 \Box C_n$ for $n \geq 6$ are both equal to $4n$. Then, consider the graphs G_{90}^6, G_{83}^6, G_{60}^6 and G_{59}^6. It is clear that G_{40}^6 is a subgraph of each of them and that G_{113}^6 is a supergraph of each of them. The result follows immediately.

□

Theorem 3.22. *Consider the cycle graph C_n, then*

1. $cr(G_{63}^6 \Box C_n) = 2n$, *for* $n \geq 4$.
2. $cr(G_{64}^6 \Box C_n) = 2n$, *for* $n \geq 6$.
3. $cr(G_{66}^6 \Box C_n) = cr(G_{70}^6 \Box C_n) = cr(G_{98}^6 \Box C_n) = 3n$, *for* $n \geq 5$.
4. $cr(G_{75}^6 \Box C_n) = 2n$, *for* $n \geq 4$.
5. $cr(G_{77}^6 \Box C_n) = 2n$, *for* $n \geq 6$.
6. $cr(G_{92}^6 \Box C_n) = 3n$, *for* $n \geq 4$.

Proof of 1. It can be seen that $G_{63}^6 \supset G_{42}^6$ and so by Lemma 3.18, $cr(G_{63}^6 \Box C_n) \geq cr(G_{42}^6 \Box C_n)$. Also, from [48], we have $cr(G_{42}^6 \Box C_n) = 2n$

for $n \geq 4$. Figure 3.20 provides a drawing procedure for $G^6_{63}\square C_n$ with $2n$ crossings and therefore $cr(G^6_{63}\square C_n) = 2n$ for $n \geq 4$.

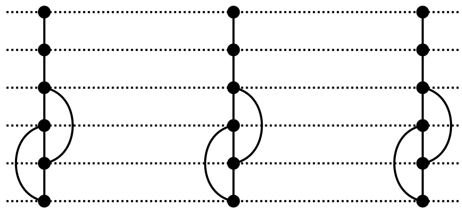


Figure 3.20: $G^6_{63}\square C_n$ drawn with $2n$ crossings. The solid lines are the edges of copies of G^6_{63} and the dashed lines are the edges of copies of C_n (which cycle around and connect to the other side).

Proof of 2. It can be seen that $G^6_{64} \supset G^6_{47}$ and so by Lemma 3.18, $cr(G^6_{64}\square C_n) \geq cr(G^6_{47}\square C_n)$. Also, from [48], we have $cr(G^6_{47}\square C_n) = 2n$ for $n \geq 6$. Figure 3.21 provides a drawing procedure for $G^6_{64}\square C_n$ with $2n$ crossings and therefore $cr(G^6_{64}\square C_n) = 2n$ for $n \geq 6$.

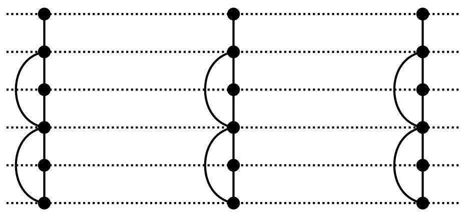


Figure 3.21: $G^6_{64}\square C_n$ drawn with $2n$ crossings. The solid lines are the edges of copies of G^6_{64} and the dashed lines are the edges of copies of C_n (which cycle around and connect to the other side).

Proof of 3. It can be seen that G^6_{98} is a supergraph of G^6_{66} and G^6_{70} and all three graphs contain G^6_{41} as a subgraph. Also, from [48], we have $cr(G^6_{41}\square C_n) = 3n$ for $n \geq 5$ which by Lemma 3.18, provides a lower bound. Figure 3.22 provides a drawing procedure for $G^6_{98}\square C_n$ with $3n$ crossings, which provides the matching upper bound.

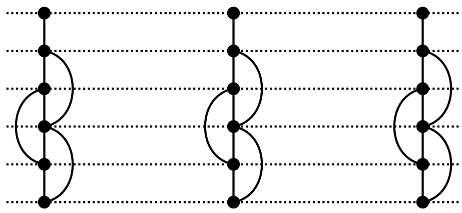


Figure 3.22: $G^6_{98} \Box C_n$ drawn with $3n$ crossings. The solid lines are the edges of copies of G^6_{98} and the dashed lines are the edges of copies of C_n (which cycle around and connect to the other side).

Proof of 4. It can be seen that $G^6_{75} \supset G^6_{49}$ and so by Lemma 3.18, $cr(G^6_{75} \Box C_n) \geq cr(G^6_{49} \Box C_n)$. Also, from [48], we have $cr(G^6_{49} \Box C_n) = 2n$ for $n \geq 4$. Figure 3.23 provides a drawing procedure for $G^6_{75} \Box C_n$ with $2n$ crossings and therefore $cr(G^6_{75} \Box C_n) = 2n$ for $n \geq 4$.

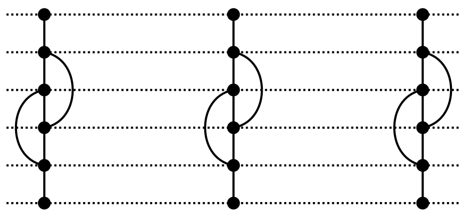


Figure 3.23: $G^6_{75} \Box C_n$ drawn with $2n$ crossings. The solid lines are the edges of copies of G^6_{75} and the dashed lines are the edges of copies of C_n (which cycle around and connect to the other side).

Proof of 5. It can be seen that $G^6_{77} \supset G^6_{53}$ and so by Lemma 3.18, $cr(G^6_{77} \Box C_n) \geq cr(G^6_{53} \Box C_n)$. Also, from [48], we have $cr(G^6_{53} \Box C_n) = 2n$ for $n \geq 6$. Figure 3.24 provides a drawing procedure for $G^6_{77} \Box C_n$ with $2n$ crossings and therefore $cr(G^6_{77} \Box C_n) = 2n$ for $n \geq 6$.

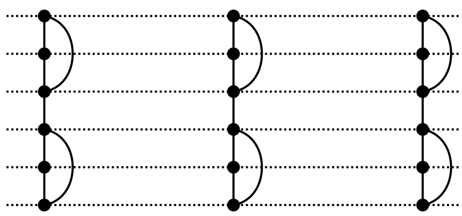


Figure 3.24: $G^6_{77} \Box C_n$ drawn with $2n$ crossings. The solid lines are the edges of copies of G^6_{77} and the dashed lines are the edges of copies of C_n (which cycle around and connect to the other side).

Proof of 6. It can be seen that $G^6_{92} \supset G^6_{67}$ and so by Lemma 3.18, $cr(G^6_{92}\Box C_n) \geq cr(G^6_{67}\Box C_n)$. Also, from [48], we have $cr(G^6_{67}\Box C_n) = 3n$ for $n \geq 4$. Figure 3.25 provides a drawing procedure for $G^6_{92}\Box C_n$ with $3n$ crossings and therefore $cr(G^6_{92}\Box C_n) = 3n$ for $n \geq 4$.

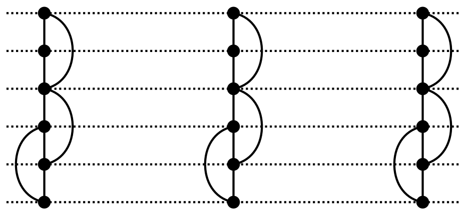


Figure 3.25: $G^6_{92}\Box C_n$ drawn with $3n$ crossings. The solid lines are the edges of copies of G^6_{92} and the dashed lines are the edges of copies of C_n (which cycle around and connect to the other side).

□

The results in Theorems 3.21 and 3.22 are stated for sufficiently large cycles. Table 3.26 presents the values for smaller cycles not covered by the theorems. Each of the values were verified using the exact crossing minimisation solver of [38], which is suitable for these small, sparse instances.

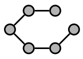
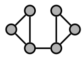
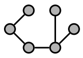
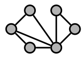
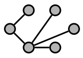
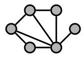
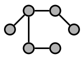
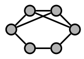
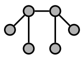
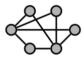
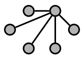
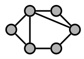
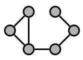
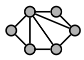
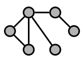
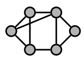
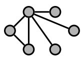
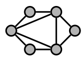
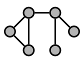
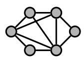
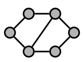
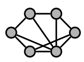
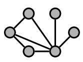
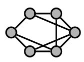
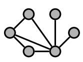
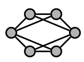
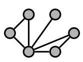
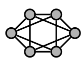
Table 3.26: Crossing numbers of Cartesian products of six vertex graphs with small cycles, which are the missing cases in Theorems 3.21 and 3.22.

i	**59**	**60**	**63**	**64**	**66**	**70**	**75**	**77**	**83**	**90**	**92**	**98**
$cr(G^6_i\Box C_3)$	8	8	6	6	7	7	6	6	10	11	9	9
$cr(G^6_i\Box C_4)$	16	16		8	12	12		8	16	16		12
$cr(G^6_i\Box C_5)$	20	20		10				10	20	20		

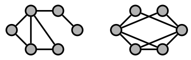
Six vertex graphs with stars

The crossing number of $G^6_i\Box S_n$ has been decided for 28 of the six vertex graphs graphs to date. These results, according to [39], are reproduced in Table 3.27.

Table 3.27: Crossing numbers of Cartesian products of graphs on six vertices with stars S_n for $n \geq 2$.

i	G_i^6	$cr(G_i^6 \Box S_n)$	i	G_i^6	$cr(G_i^6 \Box S_n)$
25		$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor$	77		$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$
26		$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+\left\lfloor\frac{n}{2}\right\rfloor$	79		$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4\left\lfloor\frac{n}{2}\right\rfloor$
27		$5\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$	80		$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4\left\lfloor\frac{n}{2}\right\rfloor$
28		$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+\left\lfloor\frac{n}{2}\right\rfloor$	85		$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n$
29		$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$	93		$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n$
31		$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4\left\lfloor\frac{n}{2}\right\rfloor$	94		$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$
43		$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+\left\lfloor\frac{n}{2}\right\rfloor$	104		$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4\left\lfloor\frac{n}{2}\right\rfloor$
47		$5\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$	111		$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor+2n$
48		$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4\left\lfloor\frac{n}{2}\right\rfloor$	120		$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+3\left\lfloor\frac{n}{2}\right\rfloor$
53		$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$	124		$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+3\left\lfloor\frac{n}{2}\right\rfloor$
59		$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$	125		$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+3\left\lfloor\frac{n}{2}\right\rfloor+2n$
61		$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n$	130		$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n$
72		$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4\left\lfloor\frac{n}{2}\right\rfloor$	137		$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n$
73		$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4\left\lfloor\frac{n}{2}\right\rfloor$	152		$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+6n$

We will now determine new results, which can be added to Table 3.27, for the following two graphs.



Theorem 3.23. *Consider the star graph S_n, then for $n \geq 2$,*

1. $cr(G^6_{62} \square S_n) = 5 \left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2 \left\lfloor \frac{n}{2} \right\rfloor$.
2. $cr(G^6_{121} \square S_n) = 6 \left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 4n$.

Proof of 1. It can be seen that $G^6_{62} \supset G^6_{27}$ and so by Lemma 3.18, $cr(G^6_{62} \square S_n) \geq cr(G^6_{27} \square S_n)$. Also, from [97], we have $cr(G^6_{27} \square S_n) = 5 \left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2 \left\lfloor \frac{n}{2} \right\rfloor$ for $n \geq 2$. Figure 3.28 provides a drawing procedure for $G^6_{62} \square S_n$ with $5 \left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2 \left\lfloor \frac{n}{2} \right\rfloor$ crossings and therefore $cr(G^6_{62} \square S_n) = 5 \left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2 \left\lfloor \frac{n}{2} \right\rfloor$ for $n \geq 2$. The number of crossings in Figure 3.28 is counted as an exercise in Appendix A.

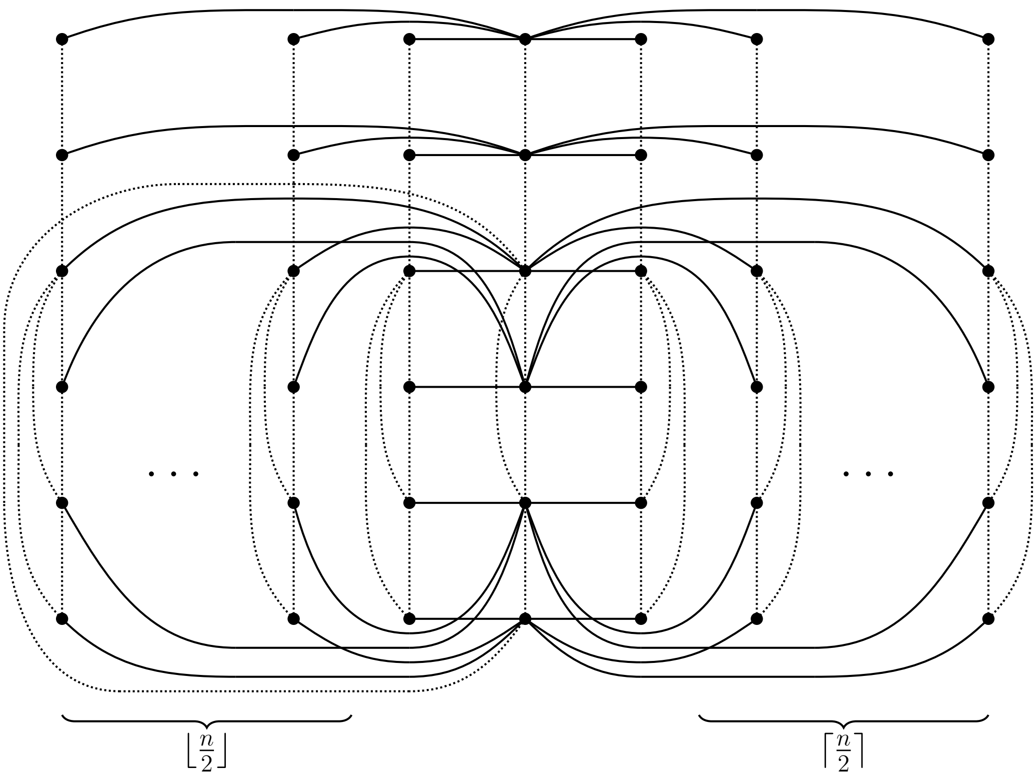


Figure 3.28: $G^6_{62} \square S_n$ drawn with $5 \left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2 \left\lfloor \frac{n}{2} \right\rfloor$ crossings. The solid lines are the edges of copies of S_n and the dashed lines are the edges of copies of G^6_{62}.

Proof of 2. It can be seen that $G^6_{121} \supset G^6_{93}$ and so by Lemma 3.18, $cr(G^6_{121}\Box S_n) \geq cr(G^6_{93}\Box S_n)$. Also, from [106], we have $cr(G^6_{93}\Box S_n) = 6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 4n$ for $n \geq 2$. Figure 3.29 provides a drawing procedure for $G^6_{121}\Box S_n$ with $6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 4n$ crossings and therefore $cr(G^6_{121}\Box S_n) = 6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 4n$ for $n \geq 2$.

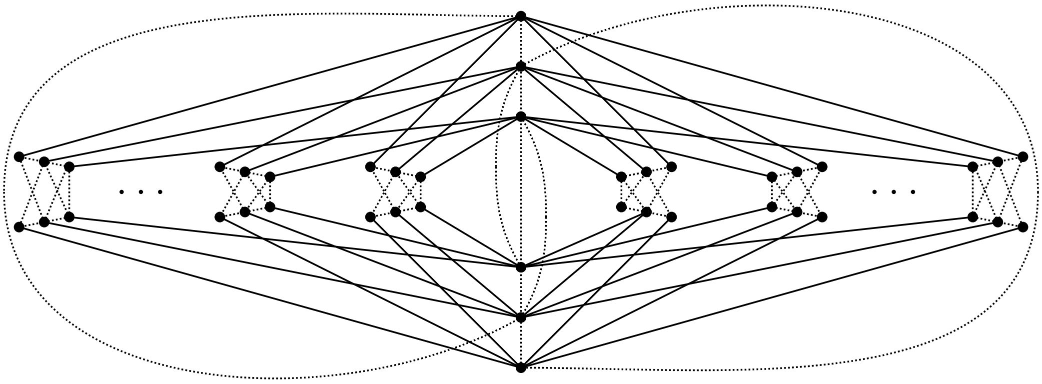


Figure 3.29: $G^6_{121}\Box S_n$ drawn with $6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 4n$ crossings. The solid lines are the edges of copies of S_n and the dashed lines are the edges of copies of G^6_{121}.

□

3.3.2 Join products involving one small graph

In addition to join products with cycles and paths, it is also common to consider the crossing number of join products of graphs with the discrete graph D_n, which is also commonly referred to as the empty graph. The discrete graph is also sometimes denoted as nK_1. To date it has been rare to consider the crossing number of join products involving arbitrarily large stars.

A number of interesting graphs can be viewed as being the result of join products. Most notably, complete multipartite graphs can be viewed as resulting from join products, in the following way. Consider the complete k-partite graph $K_{n_1,n_2,\ldots,n_k}$. Then $K_{n_1,n_2,\ldots,n_k} + D_d$ is isomorphic to the complete $(k+1)$-partite graph $K_{n_1,n_2,\ldots,n_k,d}$. Hence, a number of results for join products in fact arose from the various publications on crossing numbers of complete multipartite graphs, and vice versa.

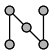
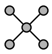
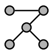
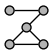
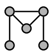
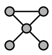
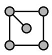
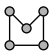
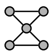
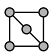
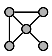
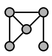
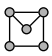
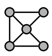
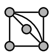
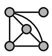
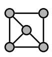
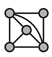
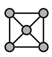
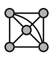
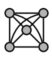
Our results in this section concern join products involving connected graphs on five and six vertices, however, we note that the crossing number of join products involving the following disconnected graphs on five and six vertices are known. $D_5 + D_n$ is the graph $K_{5,n}$ and similarly for $D_6 + D_n$. These crossing numbers are known to coincide with $Z(5,n)$ and $Z(6,n)$ respectively [89, 71]. Next, let G be the union of C_4 and one isolated vertex, then the crossing numbers of $G + D_n$, $G + P_n$ and $G + C_n$ are determined by Li in [102]. Lastly, let G be the union of G_9^4 and one isolated vertex, then the crossing number of $G + D_n$ is determined by Stăs in [132].

When referring to join products, it is common in mathematical literature to use the notation P_n to refer to the path graph on n vertices; this is contrary to the more standard usage of P_n, to refer to the path graph on $n+1$ vertices. This is because the number of vertices of each input graph is an important variable in join products, as the join product of two graphs on n_1 and n_2 vertices contains K_{n_1,n_2} as a subgraph. Hence the crossing number of the join product will inevitably contain $Z(n_1,n_2) = \left\lfloor \frac{n_1}{2} \right\rfloor \left\lfloor \frac{n_1-1}{2} \right\rfloor \left\lfloor \frac{n_2}{2} \right\rfloor \left\lfloor \frac{n_2-1}{2} \right\rfloor$. However, despite common practice in this literature, for the sake of consistency we will maintain our notation with the rest of this thesis, and use P_{n-1} to refer to the path graph on n vertices.

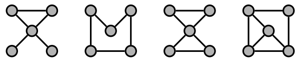
Five vertex graphs with discrete graphs, paths and cycles

We first consider the 21 connected graphs on five vertices. According to [39], the crossing numbers of their join products with discrete graphs, paths and cycles have been determined for most of the graphs and we reproduce these results in Table 3.30. It is worth noting that $n(n-1)$, a common expression in Table 3.30, is equal to $Z(5,n) + 2\left\lfloor \frac{n}{2} \right\rfloor$.

Table 3.30: Crossing numbers of joins of connected graphs on five vertices with discrete graphs, paths and cycles. Unless otherwise noted, the results for D_n are for $n \geq 1$, the results for P_{n-1} are for $n \geq 2$, and the results for C_n are for $n \geq 3$. Empty cells imply that the crossing number has not yet been determined.

i	G_i^5	$cr(G_i^5 + D_n)$	$cr(G_i^5 + P_{n-1})$	$cr(G_i^5 + C_n)$
1		$Z(5,n)$	$Z(5,n)$	$Z(5,n)+1$
2		$n(n-1)$	$n(n-1)$	$n(n-1)+2$
3		$Z(5,n)+\lfloor\frac{n}{2}\rfloor$		
4		$Z(5,n)+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+\lfloor\frac{n}{2}\rfloor+1$
5		$Z(5,n)+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+\lfloor\frac{n}{2}\rfloor$	
6			$n(n-1)$	$n(n-1)+2$
7		$Z(5,n)+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+\lfloor\frac{n}{2}\rfloor+1$	$Z(5,n)+\lfloor\frac{n}{2}\rfloor+2$
8			$Z(5,n)+1$	$Z(5,n)+2$
9			$n(n-1)$	$n(n-1)+2$
10		$Z(5,n)+n$	$Z(5,n)+n+1$	$Z(5,n)+n+3$
11		$n(n-1)$	$n(n-1)+1$	$n(n-1)+3$
12		$n(n-1)$	$n(n-1)$	
13		$Z(5,n)+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+\lfloor\frac{n}{2}\rfloor+1$	$Z(5,n)+\lfloor\frac{n}{2}\rfloor+2$
14		$n(n-1)$	$n(n-1)+1$	$n(n-1)+3$
15		$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor+2$	$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor+4$
16		$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor+1(n\geq 3)$	$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor+3$
17			$Z(5,n)+n+1$	
18		$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor+2$	$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor+4$
19		$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor+1$	$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor+4$
20		$Z(5,n)+2n$	$Z(5,n)+2n+2$	
21		$Z(5,n)+2n+\lfloor\frac{n}{2}\rfloor+1$	$Z(5,n)+2n+\lfloor\frac{n}{2}\rfloor+4$	

We will now determine new results, which can be added to Table 3.30, for the following four graphs.



G_6^5 G_8^5 G_9^5 G_{17}^5

Theorem 3.24. *Consider the discrete graph D_n, then for $n \geq 1$,*

1. $cr(G_6^5 + D_n) = cr(G_9^5 + D_n) = n(n-1)$.
2. $cr(G_8^5 + D_n) = Z(5, n)$.

Proof of 1. It can be seen that $G_9^5 \supset G_6^5 \supset G_2^5$ and so by Lemma 3.19, $cr(G_9^5+D_n) \geq cr(G_6^5+D_n) \geq cr(G_2^5+D_n)$. Also, from [82] and independently [84], we have $cr(G_2^5 + D_n) = n(n-1)$. Figure 3.31 provides a drawing procedure for $G_9^5 + D_n$ with $n(n-1)$ crossings and therefore $cr(G_9^5 + D_n) = cr(G_6^5 + D_n) = n(n-1)$.

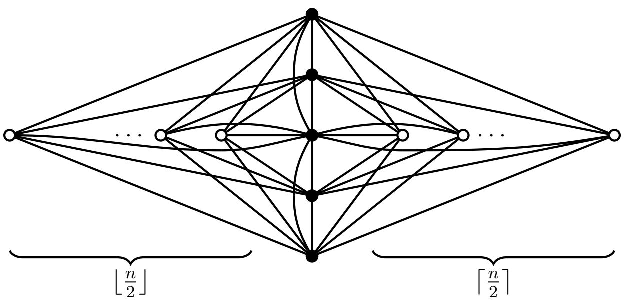


Figure 3.31: $G_9^5 + D_n$ drawn with $n(n-1)$ crossings. The black vertices are those of G_9^5 and the white vertices are those of D_n.

Proof of 2. It can be seen that $G_8^5 \supset G_1^5$ and so by Lemma 3.19, $cr(G_8^5 + D_n) \geq cr(G_1^5 + D_n)$. Also, from [89], we have $cr(G_1^5 + D_n) = Z(5, n)$. Figure 3.32 provides a drawing procedure for $G_8^5 + D_n$ with $Z(5, n)$ crossings and therefore $cr(G_8^5 + D_n) = Z(5, n)$.

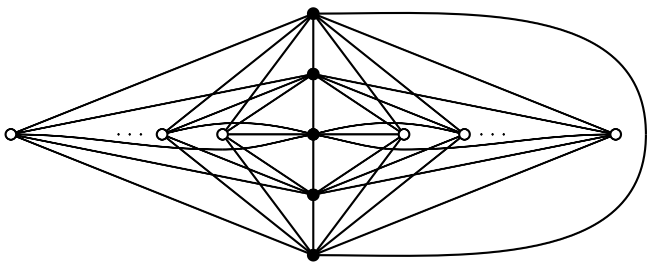


Figure 3.32: $G_8^5 + D_n$ drawn with $Z(5, n)$ crossings. The black vertices are those of G_8^5 and the white vertices are those of D_n.

□

Theorem 3.25. *Consider the cycle graph C_n, then for $n \geq 3$, $cr(G_{17}^5 + C_n) = Z(5, n) + n + 3$.*

Proof. It can be seen that $G_{17}^5 \supset G_{10}^5$ and so by Lemma 3.19, $cr(G_{17}^5 + C_n) \geq cr(G_{10}^5 + C_n)$. Also, from [146], we have $cr(G_{10}^5 + C_n) = Z(5, n) + n + 3$ for $n \geq 3$. Figure 3.33 provides a drawing procedure for $G_{17}^5 + C_n$ with $Z(5, n) + n + 3$ crossings and therefore $cr(G_{17}^5 + C_n) = Z(5, n) + n + 3$ for $n \geq 3$.

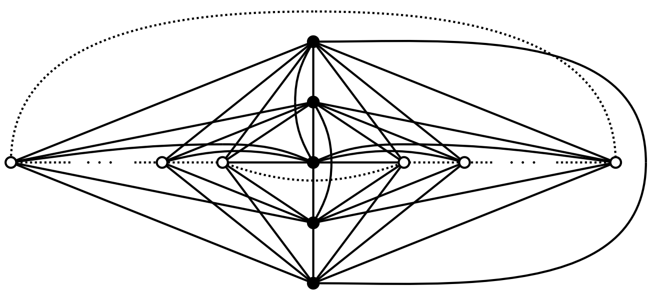


Figure 3.33: $G_{17}^5 + C_n$ drawn with $Z(5, n) + n + 3$ crossings. The black vertices are those of G_{17}^5 and the white vertices are those of C_n.

□

Six vertex graphs with discrete graphs, paths and cycles

So far, join products involving 6-vertex graphs have only been considered in-depth for connected 6-vertex graphs, and are only known for some cases.

These results, according to [39], are reproduced in Table 3.34.

Table 3.34: Crossing numbers of joins of particular graphs on six vertices with discrete graphs, paths and cycles. The results for D_n are for $n \geq 1$, the results for P_{n-1} are for $n \geq 2$, and the results for C_n are for $n \geq 3$. Empty cells imply that the crossing number has not yet been determined.

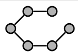
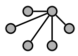
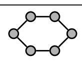
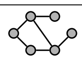
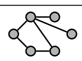
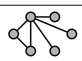
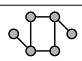
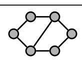
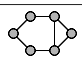
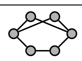
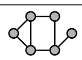
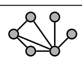
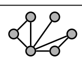
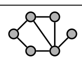
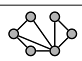
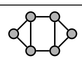
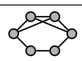
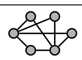
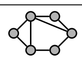
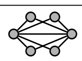
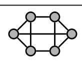
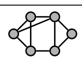
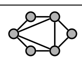
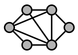
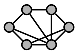
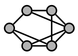
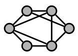
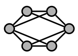
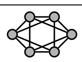
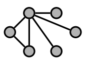
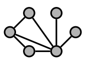
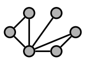
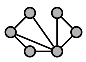
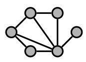
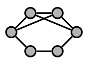
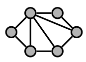
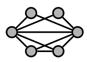
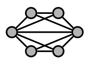
i	G_i^6	$cr(G_i^6 + D_n)$	$cr(G_i^6 + P_{n-1})$	$cr(G_i^6 + C_n)$
25		$Z(6,n)$	$Z(6,n)$	$Z(6,n)+1$
31		$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor+3$
40			$Z(6,n)+1$	$Z(6,n)+2$
44		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	
45		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$		
48			$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor$	
49		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+2$
59		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+1$	
60		$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor+2$
61		$Z(6,n)+n$	$Z(6,n)+n+1$	$Z(6,n)+n+3$
66		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$		
72			$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor$	
73			$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor$	
74		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$		
79			$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor$	
83		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+2$
85		$Z(6,n)+n$		
93		$Z(6,n)+2n$		
94		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+3$
103		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+2n$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+2n+2$	
109		$Z(6,n)+2n$	$Z(6,n)+2n+1$	$Z(6,n)+2n+3$
111		$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor+3$
120		$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor+2$	$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor+4$

Table 3.35: Continuation of Table 3.34.

i	G_i^6	$cr(G_i^6 + D_n)$	$cr(G_i^6 + P_{n-1})$	$cr(G_i^6 + C_n)$
124		$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor$		
125		$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor + 1$	
130		$Z(6,n) + 2n$		
133		$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor + 2$	
137		$Z(6,n) + 2n$		
152		$Z(6,n) + 3n$		

We will now determine new results, which can be added to Table 3.34, for the following nine graphs.

								
G_{48}^6	G_{72}^6	G_{73}^6	G_{79}^6	G_{80}^6	G_{85}^6	G_{104}^6	G_{128}^6	G_{140}^6

Theorem 3.26. *Consider the discrete graph D_n, then for $n \geq 1$,*

1. $cr(G_{48}^6 + D_n) = cr(G_{72}^6 + D_n) = cr(G_{73}^6 + D_n) = cr(G_{79}^6 + D_n) = cr(G_{80}^6 + D_n) = cr(G_{104}^6 + D_n) = Z(6,n) + 4\left\lfloor \frac{n}{2} \right\rfloor$.
2. $cr(G_{128}^6 + D_n) = cr(G_{140}^6 + D_n) = Z(6,n) + 2\left\lfloor \frac{3n}{2} \right\rfloor$.

Proof of 1. It can be seen that G_{104}^6 is a supergraph of G_{48}^6, G_{72}^6, G_{73}^6, G_{79}^6 and G_{80}^6 and all six graphs contain G_{31}^6 as a subgraph. It is shown in [110], that $cr(G_{31}^6 + D_n) = Z(6,n) + 4\left\lfloor \frac{n}{2} \right\rfloor$ which, by Lemma 3.19 provides a lower bound. Figure 3.36 provides a drawing procedure for $G_{104}^6 + D_n$ with $Z(6,n) + 4\left\lfloor \frac{n}{2} \right\rfloor$ crossings, which provides the matching upper bound.

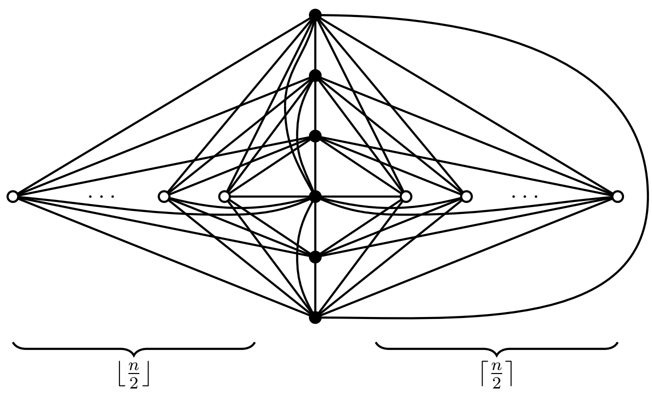


Figure 3.36: $G^6_{104} + D_n$ drawn with $Z(6,n) + 4\left\lfloor \frac{n}{2} \right\rfloor$ crossings. The black vertices are those of G^6_{104} and the white vertices are those of D_n.

Proof of 2. It can be seen that G^6_{140} is a supergraph of G^6_{128} and both graphs contain G^6_{103} as a subgraph. It is shown in [127], that $cr(G^6_{103}+D_n) = Z(6,n) + 2\left\lfloor \frac{3n}{2} \right\rfloor$ which, by Lemma 3.19 provides a lower bound. Figure 3.37 provides a drawing procedure for $G^6_{140} + D_n$ with $Z(6,n) + 2\left\lfloor \frac{3n}{2} \right\rfloor$ crossings, which provides the matching upper bound.

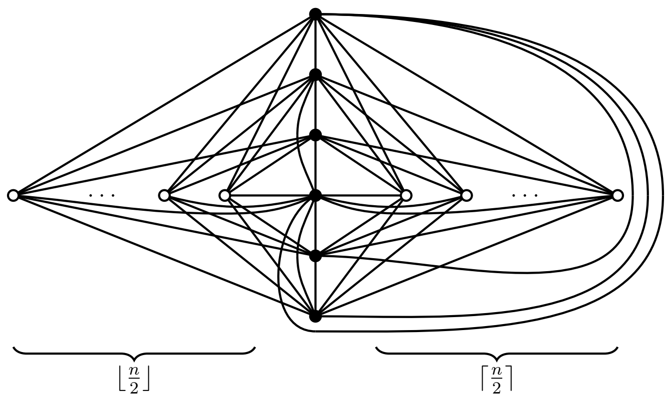


Figure 3.37: $G^6_{140} + D_n$ drawn with $Z(6,n) + 2\left\lfloor \frac{3n}{2} \right\rfloor$ crossings. The black vertices are those of G^6_{140} and the white vertices are those of D_n.

□

Theorem 3.27. *Consider the path graph P_n, then for $n \geq 2$, $cr(G^6_{85}+P_n) = Z(6,n) + n + 1$.*

Proof. It can be seen that $G^6_{85} \supset G^6_{61}$ and so by Lemma 3.19, $cr(G^6_{85}+P_n) \geq cr(G^6_{61}+P_n)$. Also, from [98], we have $cr(G^6_{61}+P_n) = Z(6,n)+n+1$ for $n \geq 2$. Figure 3.38 provides a drawing procedure for $G^6_{85}+P_n$ with $Z(6,n)+n+1$ crossings and therefore $cr(G^6_{85}+P_n) = Z(6,n)+n+1$ for $n \geq 2$.

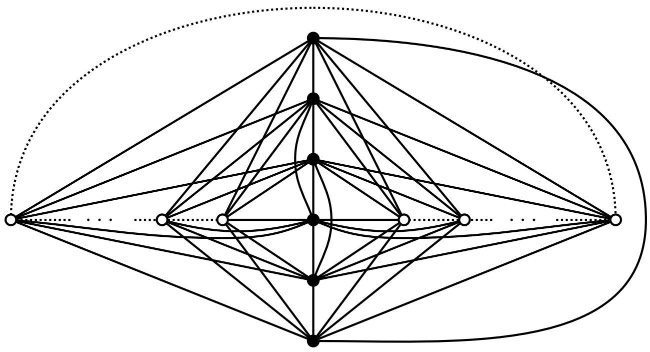


Figure 3.38: $G^6_{85}+P_n$ drawn with $Z(6,n)+n+1$ crossings. The black vertices are those of G^6_{85} and the white vertices are those of P_n.

□

Theorem 3.28. *Consider the cycle graph C_n, then for $n \geq 3$,*

1. $cr(G^6_{48}+C_n) = cr(G^6_{72}+C_n) = cr(G^6_{73}+C_n) = cr(G^6_{79}+C_n) = Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor+3$.
2. $cr(G^6_{85}+C_n) = Z(6,n)+n+3$.

Proof of 1. It can be seen that G^6_{79} is a supergraph of G^6_{48}, G^6_{72}, G^6_{73} and all four graphs contain G^6_{31} as a subgraph. It is shown in [134], that $cr(G^6_{31}+C_n) = Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor+3$ for $n \geq 3$ which, by Lemma 3.19 provides a lower bound. Figure 3.39 provides a drawing procedure for $G^6_{79}+C_n$ with $Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor+3$ crossings, which provides the matching upper bound.

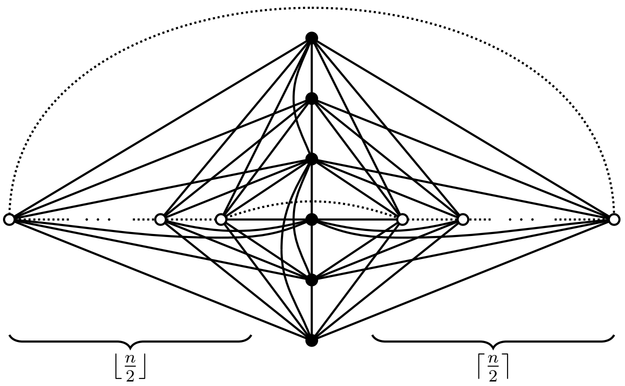


Figure 3.39: $G^6_{79} + C_n$ drawn with $Z(6,n) + 4\left\lfloor \frac{n}{2} \right\rfloor + 3$ crossings. The black vertices are those of G^6_{79} and the white vertices are those of C_n.

Proof of 2. It can be seen that $G^6_{85} \supset G^6_{61}$ and so by Lemma 3.19, $cr(G^6_{85} + C_n) \geq cr(G^6_{61} + C_n)$. Also, from [98], we have $cr(G^6_{61} + C_n) = Z(6,n) + n + 3$ for $n \geq 3$. Figure 3.40 provides a drawing procedure for $G^6_{85} + C_n$ with $Z(6,n) + n + 3$ crossings and therefore $cr(G^6_{85} + C_n) = Z(6,n) + n + 3$ for $n \geq 3$.

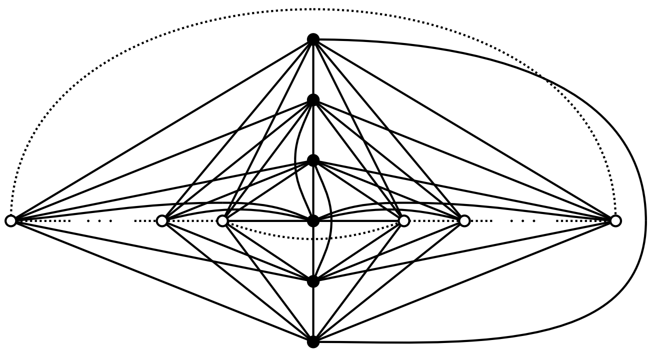


Figure 3.40: $G^6_{85} + C_n$ drawn with $Z(6,n) + n + 3$ crossings. The black vertices are those of G^6_{85} and the white vertices are those of C_n.

□

Given the existing catalogues of known results, along with the new results determined in this section, we can now identify all of the remaining graph families resulting from the five and six vertex graphs for which results are yet to be determined. Specifically, there are 607 such cases. Much like when proving results involving the Sunlet graphs in Section 3.2, it can be valuable to predict, in advance, what the crossing numbers are likely to be. To that end, we decided to use Quickcross to help predict the crossing numbers. First, we considered all cases where the result is known. For each family, we ran the first few instances with Quickcross, and used the results to make a prediction about the formula for its crossing numbers. In every case where the formula for the crossings numbers are known, it coincided with the predicted formula from Quickcross. Given that Quickcross was able to accurately predict all of the known cases, we then turned to the remaining cases. The full list of predicted results for the graph families resulting from five and six vertex graphs are displayed and discussed further in Appendix C.

To conclude this chapter, we now make a brief point about benchmarking. In literature, there are a few benchmark sets of instances for the crossing number problem. For example, several of these were considered in Section 2.5. However, for most of these, the actual crossing numbers are not known. It is only the KnownCR instances, originally collected by Gutwenger [65] in Section 4.3.2 of his thesis, where the correct values are known and have been proved. The recent comprehensive survey of known results, along with the additional results form this chapter, could be used to significantly expand upon the KnownCR set for use in future research. The consideration of all known results in order to produce a broad benchmark set with known crossing numbers is a topic for future research.

Chapter 4

New bounds and conjectures relating to crossing numbers

Throughout Chapter 3, we were able to take advantage of Quickcross to directly aid in the problems we considered. In this chapter, we again take advantage of Quickcross, however, it will now only serve as a guide in our investigations instead of a direct aid. This is because the types of graphs that we investigate in this chapter present some algorithmic difficulties in terms of tractability and solution quality. These difficulties have also contributed to the lack of exact results for many of the graphs considered in this chapter. Consequently, the power of Quickcross as a prediction tool for the crossing number is limited, and it becomes unreliable even when the graphs are of moderate size. Nonetheless, it is still useful for the small instances, and we take advantage of this here.

This chapter is laid out as follows. First, in Section 4.1, we formalise an observation that was made during our investigations in Section 3.3. The observation was that the crossing number of a Cartesian product of a fixed graph with an arbitrarily large cycle appeared to always obey a simple formula. The formula arises by considering the crossing number of a much smaller and simpler graph. We conjecture that the crossing number of a

Cartesian product of a fixed graph with an appropriately large cycle always obeys this formula and we also demonstrate that all known proved cases agree with our conjecture. This work can be viewed as a narrower and alternative approach to the general theory of tiles, which was initially developed by Pinontoan and Richter in [115], and then further in [116] and [49].

In Section 4.2, we study the crossing numbers of a family of dense graphs, which has only been determined in literature for a few small cases. Specifically, we study the two-parameter family of graphs arising from the join product of a complete graph and a discrete graph. We begin by determining lower bounds on the crossing numbers by adapting a classical counting argument to this family. We then take advantage of properties of cylinder drawings of the complete graph to determine an upper bound on the crossing numbers of this family. Although it is not obvious from the drawings that the upper bound would be tight, values obtained from Quickcross seem to suggest that they are, and we conclude this section with a conjecture that this is the case.

Next, in Section 4.3, we study the well-known family of generalised Petersen graphs with parameters n and k. Specifically, we focus on the smallest cases for which the crossing numbers are unknown. We show that Quickcross is able to obtain results which coincide with recent conjectures about the crossing numbers for the case where $k = 4$ and n is arbitrary. Motivated by this, we use Quickcross to consider the case when $k = 5$ and n is arbitrary, for which there are no conjectures about the equality of crossing numbers in literature and provide the drawing procedures which give these upper bounds. We determine new upper bounds, which we conjecture are equal to the crossing numbers for this case. We were able to discover these drawing procedures by running Quickcross many times, storing the drawings it produced and then identifying which of these drawings have properties which may be generalised.

Next, in Section 4.4, we consider the n-cube, a graph whose vertices and edges correspond to those of the n-dimensional cube. Research on the crossing numbers of n-cubes has a fascinating history and even determining upper bounds remains a notoriously difficult problem. One reason for this is that the number of vertices of the n-cube increases exponentially with n. Quickcross is able to obtain a solution to the 7-cube and 8-cube with fewer crossings than a long-conjectured upper bound from the 1970's. The fact that the upper bound was not tight was recently discovered by Yang et al. [145] who produced a constructive procedure for drawing the n-cube with fewer crossings. However, this is the first time a heuristic approach has discovered such a drawing.

Lastly, in Section 4.5, we make some preliminary observations about a family of graphs, the Sheehan graphs, whose crossing number has not yet been investigated. We observe that the the crossing number of these graphs is of the same order as the complete graphs. We also observe that the Sheehan graphs provide significantly difficult instances for the tested crossing minimisation heuristics. We also demonstrate a formula for the number of crossings, for which our best found solutions follow very closely.

4.1 Cartesian products involving cycles

A common line of research into crossing numbers has been to consider infinite families of graphs which result from graph products. As discussed in Section 3.2, the earliest such published result was due to Harary, Kainen and Schwenk [75] who conjectured the following.

Conjecture 4.1. *For $m \geq n \geq 3$, consider the cycles C_n and C_m. Then,*

$$cr(C_n \square C_m) = (n-2)m.$$

Despite an enormous amount of effort, this conjecture (in its original form) remains unproven. Indeed, it appears that determining the crossing numbers of graph families resulting from Cartesian products involving cycles is, in general, a difficult task. In [39], an exhaustive list of graphs and graph families with known crossing numbers was given, including Cartesian products involving small fixed graphs with arbitrarily large paths, cycles, and stars. Of this set, it was the Cartesian products involving cycles that had the fewest proved results.

In this section, we propose a conjecture for the crossing numbers of Cartesian products involving arbitrarily large cycles. In particular, if the new conjecture can be proved, we will show that Conjecture 4.1 follows as well. We will demonstrate that all known proved crossing number results involving Cartesian products of cycles obey this new conjecture. Furthermore, we will use Quickcross to do an expansive experimental analysis of Cartesian products involving cycles to provide evidence that the new conjecture is valid.

For any given graph G, we define a related multigraph $\widehat{G}$ which will be useful in the upcoming discussion.

Definition 4.2. Consider a graph G containing n vertices. Then $\widehat{G}$ is formed in the following way. Begin with the union of G, and two vertices v_1 and v_2. For every vertex v in G, add edges (v, v_1) and (v, v_2) to $\widehat{G}$. Finally, add n edges (v_1, v_2) (so that the result is a set of multiedges of cardinality n).

An example of a graph G and its related $\widehat{G}$ is displayed in Figure 4.1.

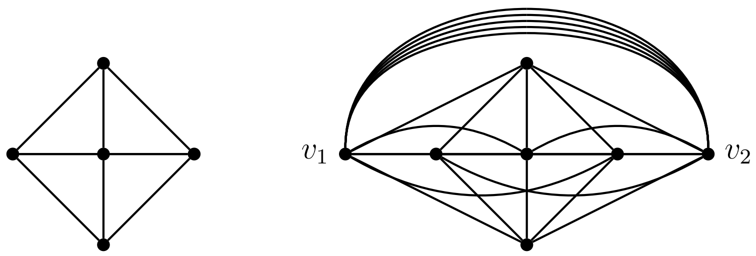


Figure 4.1: On the left, a graph G and on the right, the corresponding graph $\widehat{G}$.

In what follows, we will provide evidence that the following conjecture is true.

Conjecture 4.3. *For a given graph G containing n vertices, define $p := cr(\widehat{G})$. Then, there exists an integer $q_G \geq 0$ such that, for all $m > n$,*

$$cr(G \square C_m) = \begin{cases} mp & \text{if } m \text{ is even,} \\ mp + q_G & \text{if } m \text{ is odd.} \end{cases}$$

The rationale for believing Conjecture 4.3 is as follows. The graph $G \square C_m$ can be thought of as consisting of m individual copies of G, plus some edges corresponding to C_m which join the copies together. Then, suppose we have an optimal drawing of $\widehat{G}$. It is obviously undesirable to cross the multiedge since this would introduce n crossings, so we assume that the crossings occur elsewhere (this will be proved shortly in Lemma 4.4). Then, if those multiedges are deleted, and v_1 and v_2 are replaced with copies of G, what results is one small part of $G \square C_m$. Clearly then, the p crossings in an optimal drawing of $\widehat{G}$ correspond to the crossings on an individual copy of G, plus any extra crossings introduced by the edges leaving that copy to go to its neighbouring copies. This can then be replicated for each copy of G, resulting in mp crossings. The only remaining question is if there are any extra crossings introduced as the dangling edges are joined together to form $G \square C_m$. We will prove in Theorem 4.6 that when m is even, this can be achieved without introducing any new crossings, and when m is odd, crossings may be introduced, but only between at most one pair of neighbouring copies of G. Finally, we will then provide empirical evidence that this does, indeed, lead to an optimal drawing of the graph.

We begin with the following lemma.

Lemma 4.4. *There exists an optimal drawing of $\widehat{G}$ such that none of the multiedges (v_1, v_2) are involved in any crossings.*

Proof. Consider an optimal drawing $\widehat{D}$ of $\widehat{G}$. Clearly the number of crossings on each multiedge connecting v_1 and v_2 must be the same, otherwise we may reroute them to be arbitrarily close to the multiedge with the fewest crossings and obtain a contradiction to the optimality. The same argument also implies that $\widehat{D}$ can be modified, without increasing the number of crossings, so that all multiedges between v_1 and v_2 lie arbitrarily close to each other and cross the exact same set of edges. Call the resulting drawing $\widehat{D}'$. Now, in $\widehat{D}'$, there are n edges emanating from v_1 which connect to vertices of G, and there are n multiedges emanating from v_1 which connect to v_2. The modifications to produce $\widehat{D}'$ ensure that the cyclic ordering of edges emanating from v_1 are partitioned so that the n multiedges between v_1 and v_2 occur consecutively and then the n edges connecting v_1 to vertices of G occur afterwards. This situation is displayed in Figure 4.2 and the same property holds for v_2. This permits $\widehat{D}'$ to be further modified, without increasing the number of crossings, so that v_1 and v_2 lie arbitrarily close to each other. To do this, consider any edge e which crosses the multiedges between v_1 and v_2. Then v_1 can be moved to the opposite side of e. This removes at least n crossings from the multiedges and then the edges between v_1 and the vertices of G can be drawn so that they cross e where the multiedges used to cross. The partitioning of the edges around v_1 ensures that the total number of crossings does not increase. This process is illustrated in Figure 4.3 (a) and (b). Thus we have obtained an optimal drawing of $\widehat{G}$ with no crossings on the multiedges (v_1, v_2). □

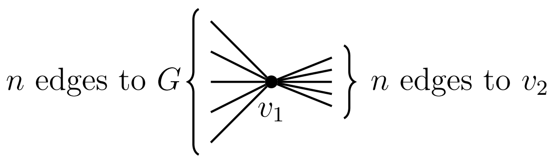


Figure 4.2: The cyclic ordering of edges around v_1 in $\widehat{D}'$.

Corollary 4.5. *Any optimal drawing $\widehat{D}$ satisfying Lemma 4.4 can be drawn so that v_1 and v_2 lie in the unbounded region.*

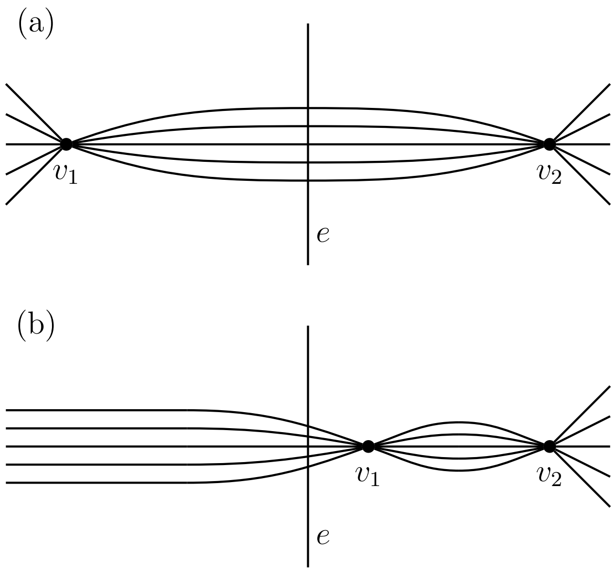


Figure 4.3: In (a), the multiedges between v_1 and v_2 are crossed by e. In (b), v_1 has been moved to the other side of e without increasing the total number of crossings.

Proof. Consider the planarisation of $\widehat{D}$. Since there are no crossings on the multiedges (v_1, v_2), it is clear that there exists a face in this planarisation that contains both v_1 and v_2. Then, it is sufficient to make this face the unbounded face. $\square$

Lemma 4.4 and Corollary 4.5 lead to the following upper bound on $cr(G\Box C_m)$.

Theorem 4.6. *For a given graph G containing n vertices, define $p := cr(\widehat{G})$. Then, there exists an integer $q_G \geq 0$ such that, for all $m > n$,*

$$cr(G\Box C_m) \leq \begin{cases} mp & \textit{if } m \textit{ is even}, \\ mp + q_G & \textit{if } m \textit{ is odd}. \end{cases}$$

Proof. We shall construct a drawing of $G\Box C_m$ which meets the upper bound. The approach will be to start with m disjoint drawings of G, and then join these together in such a way as to produce $G\Box C_m$. First, consider the graph $\widehat{G}$, and let $\widehat{D}$ be an optimal drawing of $\widehat{G}$ such that there are no crossings on the multiedges (v_1, v_2); from Lemma 4.4, we know that such a drawing must exist, and from Corollary 4.5 we can draw this drawing so that v_1 and v_2 lie in the unbounded region. Then, delete the multiedges (v_1, v_2) from $\widehat{D}$.

Next, we create m disjoint drawings of G, which we denote by D_i for $i = 1, 2, \ldots, m$. For the drawings with odd label, we simply set $D_i = \widehat{D}$, and for the drawings with even label, we set D_i to be equal to the mirror image of $\widehat{D}$. For each D_i, let $v_{1,i}$ and $v_{2,i}$ denote the vertices v_1 and v_2 from the definition of $\widehat{G}$. We then arrange these m drawings in a circle in the order of their indices. If m is even, it is clear that every drawing is the mirror image of both of its neighbours. If m is odd, then there is exactly one pair of neighbours which are drawn identically. Without loss of generality, let this pair be D_1 and D_m. These situations are displayed, in detail, in Figure 4.4.

We will first handle the case when m is even. For each drawing D_{2i} with even label, consider its neighbour D_{2i-1}. Since they are reflections, the cyclic permutation of edges emanating from $v_{2,2i}$ is exactly the inverse of those around $v_{2,2i-1}$. This implies that we can modify the drawing so that the vertices $v_{2,2i}$ and $v_{2,2i-1}$ are deleted, and then each vertex of D_{2i} is connected, by a new edge, to its corresponding vertex in D_{2i-1}, by simply following the previously deleted edges. An example can be seen in Figure 4.5. It can be easily seen that doing so does not alter the number of crossings. Then, an analogous modification can be made for drawings D_{2i} and D_{2i+1} on vertices $v_{1,2i}$ and $v_{1,2i+1}$ without altering the number of crossings. Applying this modification to all drawings with even label produces a drawing of the graph $G \square C_m$. Since the number of crossings has not changed, it is simply equal to mp.

Next, we handle the case when m is odd. The argument from the previous paragraph can be applied to all drawings D_{2i} with even label here as well, and the number of crossings remains unchanged at mp. Then, all that remains is to modify the drawings of D_1 and D_m to obtain a drawing of $G \square C_m$. Vertices $v_{1,1}$ and $v_{2,m}$ still exist in the drawing at this stage. We can then

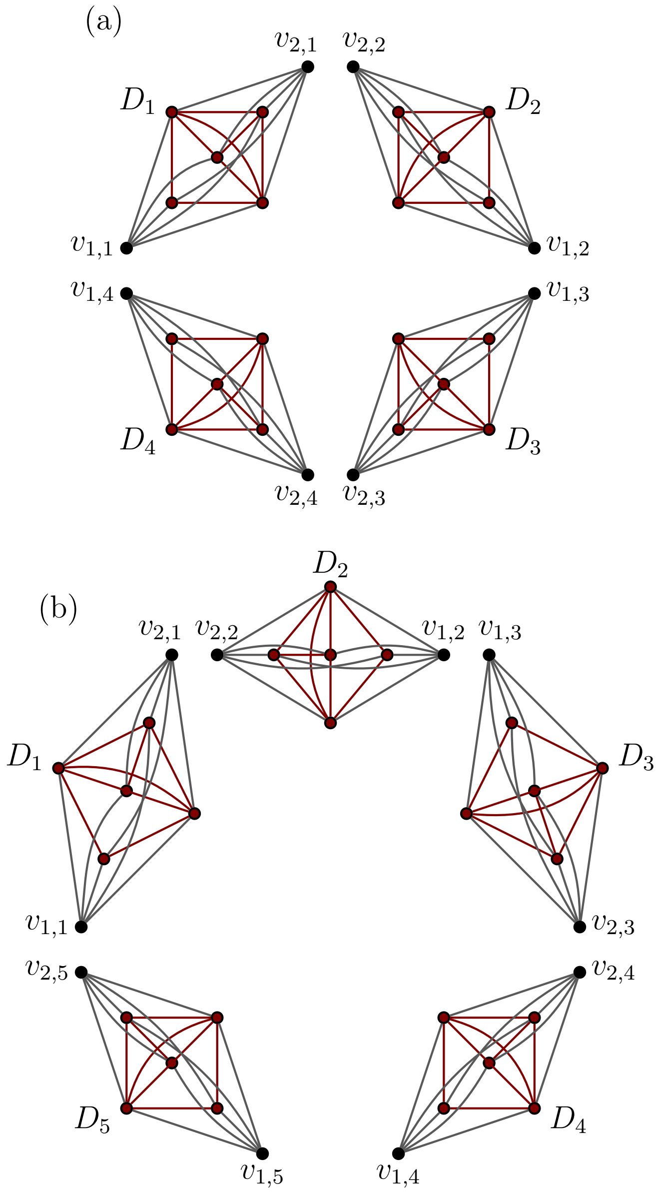


Figure 4.4: In (a), each D_i is a mirror image of D_{i-1}. In (b), each D_i is a mirror image of D_{i-1}, with the exception of D_1 which is identical to D_5. The edges and vertices of G inside each D_i are highlighted red to emphasise that they are mirror imaged.

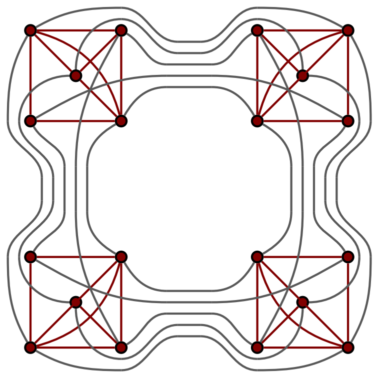


Figure 4.5: The modifications to produce a drawing of $G\Box C_m$, when m is even.

simply delete these vertices, and each vertex of D_1 can be connected, by a new edge, to its corresponding vertex in D_m. However, in this case, we may need to increase the number of crossings to do so. Suppose that this is done in an optimal way (so as to introduce the least new crossings). Then we set q_G to be the number of new crossings introduced in this step. The result is a drawing of $G\Box C_m$ which contains $mp + q_G$ crossings. □

Given that Theorem 4.6 provides a valid upper bound for Cartesian products involving cycles, we now revisit Conjecture 4.3 which claims that the crossing number coincides with this upper bound. We now provide evidence to support Conjecture 4.3.

We first consider all proved results for crossing numbers of Cartesian products involving cycles. In a recent survey [39] of graph families with known crossing numbers, an exhaustive list of Cartesian products involving cycles with known crossing numbers was provided. In particular, the following results are known.

Theorem 4.7. *Consider the path P_n for $n \geq 1$ and the cycle C_m for $m \geq 3$. Then,*

$$cr(P_n\Box C_m) = 0.$$

Theorem 4.8 ([121, 19, 96, 118, 6, 62])**.** *Consider the cycles C_n and C_m for*

$3 \leq m \leq n$. *Then, if either* $m \leq 7$, *or* $m(m+1) \leq n$,

$$cr(C_n \square C_m) = (m-2)n.$$

Theorem 4.9 (Jendrol and Ščerbová, 1982 [85]). *Consider the star* S_3, *and the cycle* C_m *for* $m \geq 3$. *Then,* $cr(S_3 \square C_3) = 1$, $cr(S_3 \square C_4) = 2$, $cr(S_3 \square C_5) = 4$, *and*

$$cr(S_3 \square C_m) = m, \quad n \geq 6.$$

Theorem 4.10 (Klešč, 1991 [90]). *Consider the star* S_4, *and the cycle* C_m *for* $m \geq 3$. *Then,* $cr(S_4 \square C_3) = 2$, $cr(S_4 \square C_4) = 4$, $cr(S_4 \square C_5) = 8$, *and*

$$cr(S_4 \square C_m) = 2m, \quad n \geq 6.$$

Theorem 4.11 (Zheng et al., 2008 [148]). *Consider the complete graph* K_n *for* $n \leq 7$, *and the cycle* C_m *for* $m \geq 3$. *Then,*

$$cr(K_n \square C_m) = \frac{1}{4} \left\lfloor \frac{n+2}{2} \right\rfloor \left\lfloor \frac{n+1}{2} \right\rfloor \left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor m. \tag{4.1}$$

Also, equation (4.1) holds if $n = 8, 9, 10$ *and* m *is even.*

Finally, there are a number of graph families resulting from the Cartesian product of small fixed graphs with arbitrarily large cycles. These were discussed in detail in Section 3.3, and in Table 4.6, we reproduce the list, given in [39], of all such known results, adding in the twelve new results proved in Section 3.3.

Recall that Conjecture 4.3 makes two claims. First, that $cr(G \square C_m)$ is linear in m, for $m \geq |G|$, and second that the leading coefficient can be determined by finding the crossing number of $\widehat{G}$. It is clear from the above known results that the crossing number is, indeed, linear in m in all cases. Perhaps interestingly, the constant term $q_G = 0$ in all of those cases other than the case $G^5_{16} \square C_m$ where $q_G = 1$. Then, to confirm that these results all agree with Conjecture 4.3, we need to find $cr(\widehat{G})$ for the various graphs.

Table 4.6: For graphs G such that $cr(G\square C_m)$ is known, we display their crossing numbers. All results are correct for $m \geq 1$.

id	G	$cr(G\square C_m)$	id	G	$cr(G\square C_m)$	id	G	$cr(G\square C_m)$
G_1^3		0	G_{11}^5		$3m$	G_{59}^6		$4m$
G_2^3		m	G_{12}^5		$2m$	G_{60}^6		$4m$
G_1^4		m	G_{13}^5		$3m$	G_{63}^6		$2m$
G_2^4		0	G_{14}^5		$3m$	G_{64}^6		$2m$
G_3^4		m	G_{16}^5		$2(m + \lfloor \frac{m+1}{2} \rfloor)$	G_{66}^6		$3m$
G_4^4		$2m$	G_{21}^5		$9m$	G_{67}^6		$3m$
G_5^4		$2m$	G_{25}^6		0	G_{70}^6		$3m$
G_6^4		$3m$	G_{40}^6		$4m$	G_{75}^6		$2m$
G_1^5		0	G_{41}^6		$3m$	G_{77}^6		$2m$
G_2^5		$2m$	G_{42}^6		$2m$	G_{78}^6		$3m$
G_3^5		m	G_{43}^6		m	G_{83}^6		$4m$
G_4^5		m	G_{44}^6		$2m$	G_{90}^6		$4m$
G_5^5		m	G_{46}^6		m	G_{92}^6		$3m$
G_6^5		$2m$	G_{47}^6		$2m$	G_{98}^6		$3m$
G_7^5		$2m$	G_{49}^6		$2m$	G_{113}^6		$4m$
G_8^5		$3m$	G_{53}^6		$2m$	G_{156}^6		$18m$
G_9^5		$2m$	G_{54}^6		$2m$	C_7		$5m$

Theorem 4.12. *The following three items are true for $m \geq 1$.*

1. $cr(\widehat{P}_m) = 0$,
2. $cr(\widehat{S}_m) = \lfloor \frac{m}{2} \rfloor \lfloor \frac{m-1}{2} \rfloor$,
3. $cr(\widehat{C}_m) = m - 2$.

Proof. We first consider items 1 and 2. It is obvious that $\widehat{P}_m$ is planar, as can be seen in Figure 4.7 (a) satisfying item 1. It is also clear that $\widehat{S}_m$ is isomorphic to the graph $K_{1,1,1,m}$ with one of the edges connecting the partitions of cardinality one replaced by a multiedge of cardinality $m+1$. If there exists an optimal drawing of $K_{1,1,1,m}$ such that one of the edges connecting the partitions of cardinality one is not crossed, then it is clear that the crossing numbers of $\widehat{S}_m$ and $K_{1,1,1,m}$ coincide. Figure 4.7 (b) displays such a drawing of $K_{1,1,1,m}$. Then, it was shown in [117] that $cr(K_{1,1,1,m}) = Z(3, n) = \lfloor \frac{m}{2} \rfloor \lfloor \frac{m-1}{2} \rfloor$, satisfying item 2.

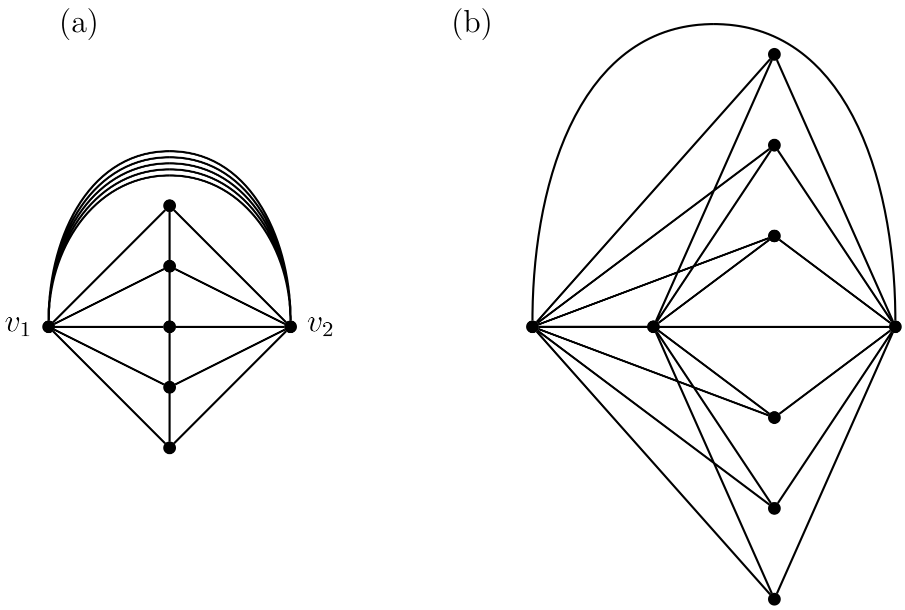


Figure 4.7: In (a), a planar drawing of $\widehat{P}_m$. In (b), a procedure for drawing $K_{1,1,1,m}$ optimally such that one of the edges connecting the partitions of cardinality one is not crossed.

We now turn our attention to item 3. In $\widehat{C}_m$, label the vertices of C_m in a cyclic fashion as $u_1, u_2, \ldots, u_m$. First, we claim that in any optimal drawing

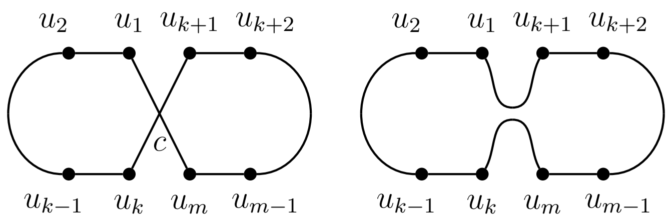


Figure 4.8: On the left, c is a crossing between two edges of C_m. On the right, the modification removes c and reduces the total number of crossings by one.

of $\widehat{C}_m$, there are no crossings between any pair of edges of C_m. To show this, we follow of argument of Klešč et al. in [95]. Assume the contrary and let c be a crossing on the edges of C_m. Then the crossing c partitions the vertices of C in two sets as $u_1, u_2, \ldots, u_k$ and $u_{k+1}, u_{k+2}, \ldots, u_m$. But then we may redraw C_m as in Figure 4.8 so that the cyclic labelling of C_m is now $u_1, u_2, \ldots, u_k, u_m, u_{m-1}, \ldots, u_{k+1}$ and the crossing c is removed. Since each vertex u_i is linked to v_1 and v_2, the other edges in the graph do not need to be redrawn. Hence, the number of crossings has reduced by one, contradicting the optimality. This claim ensures that in any optimal drawing of $\widehat{C}_m$, either the crossings occur on the multiedges, or they occur on edges of the form (v_1, u_i) or (v_2, u_j).

Next, we inductively prove that $cr(\widehat{C}_m) = m - 2$. The graph $\widehat{C}_3$ contains K_5 as a subgraph and thus $cr(\widehat{C}_3) \geq 1$. This, along with the upper bound from Theorem 4.6 implies that $cr(\widehat{C}_3) = 1$. Now, assume that $\widehat{C}_m = m - 2$ holds for all $m = 3, 4, \ldots, k-1$ but that $\widehat{C}_k < k - 2$. Consider an optimal drawing of $\widehat{C}_k$ with no crossings on any of the multiedges. By Lemma 4.4, such a drawing exists. Then, by the earlier claim, there must exist at least one pair of edges $e = (v_1, u_i)$ and $f = (v_2, u_j)$ such that there is a crossing on either e or f. Deleting edges e and f produces a graph which is homeomorphic to $\widehat{C}_{k-1}$ and has at most $k - 3 - 1 = k - 4$ crossings, which contradicts our inductive assumption. Hence we obtain that $cr(\widehat{C}_m) = m - 2$ which proves item 3. $\square$

Theorem 4.12 verifies Conjecture 4.3 for the results in Theorems 4.7-4.11. For the result in 4.11, it is worth noting that the graph $\widehat{K}_m$ is isomorphic to K_{m+2} with one of the edges replaced by a multiedge of cardinality m. Note that all complete graphs are edge-transitive, so it doesn't matter which edge is replaced. If there exists an optimal drawing of K_{m+2} such that there is at least one edge drawn with no crossings, then the crossing numbers of $\widehat{K}_m$ and K_{m+2} coincide. It is currently an open question whether there always exists optimal drawings of the complete graph such that there is an edge drawn with no crossings on it. However, for the cases where the crossing number of K_n is known, such drawings are known to exist. Hence, $cr(\widehat{K}_m) = cr(K_{m+2})$ for $m \leq 10$ and so these cases also support Conjecture 4.3.

Finally, we consider the various results involving the Cartesian product of a small fixed graph with arbitrarily large cycles. For each of these small fixed graphs G, we used [38] to compute $cr(\widehat{G})$. Their crossing numbers are displayed in Table 4.9, and it is a simple exercise to check that they coincide with the coefficients in Table 4.6.

Given that all known results for crossing numbers of Cartesian products involving cycles agree with Conjecture 4.3, we conducted an additional, thorough experimental exercise to provide further evidence. Recall that in Section 3.3, we used Quickcross to predict the crossing numbers of Cartesian products of six-vertex graphs with arbitrarily large cycles, paths and stars; these are included in Appendix C. For the special case of cycles, we extended this exercise to all small fixed graphs of orders $n = 5, 6, 7, 8$ and also used Quickcross to predict the crossing number of $\widehat{G}$ in each case. Note that this set includes 12,103 graphs, so we do not list the results here. In every single tested case, the crossing numbers predicted by Quickcross agreed with Conjecture 4.3. Although this experiment does not conclusively determine the various crossing numbers, it nonetheless provides additional empirical evidence that Conjecture 4.3 is true.

Table 4.9: For graphs G such that $cr(G\square C_m)$ is known, we display the crossing numbers of $\widehat{G}$.

id	G	$cr(\widehat{G})$	id	G	$cr(\widehat{G})$	id	G	$cr(\widehat{G})$
G^3_1		0	G^5_{11}		3	G^6_{59}		4
G^3_2		1	G^5_{12}		2	G^6_{60}		4
G^4_1		1	G^5_{13}		3	G^6_{63}		2
G^4_2		0	G^5_{14}		3	G^6_{64}		2
G^4_3		1	G^5_{16}		2	G^6_{66}		3
G^4_4		2	G^5_{21}		9	G^6_{67}		3
G^4_5		2	G^6_{25}		0	G^6_{70}		3
G^4_6		3	G^6_{40}		4	G^6_{75}		2
G^5_1		0	G^6_{41}		3	G^6_{77}		2
G^5_2		2	G^6_{42}		2	G^6_{78}		3
G^5_3		1	G^6_{43}		1	G^6_{83}		4
G^5_4		1	G^6_{44}		2	G^6_{90}		4
G^5_5		1	G^6_{46}		1	G^6_{92}		3
G^5_6		2	G^6_{47}		2	G^6_{98}		3
G^5_7		2	G^6_{49}		2	G^6_{113}		4
G^5_8		3	G^6_{53}		2	G^6_{156}		18
G^5_9		2	G^6_{54}		2	C_7		5

4.2 Join product of a complete graph and a discrete graph

Let K_n be the complete graph on n vertices and let D_d be the discrete graph on d vertices, that is, the graph consisting of d isolated vertices. In the following, we will consider the join product $K_n + D_d$. For example, $K_3 + D_3$ is shown in Figure 4.10.

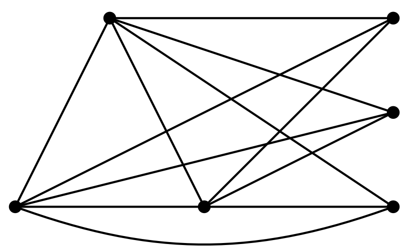


Figure 4.10: A drawing of $K_3 + D_3$.

The graph $K_n + D_d$ is isomorphic to the complete multipartite graph $K_{d,1,1,\ldots,1}$, where there are n partitions of cardinality one. Some crossing numbers for this family are known and for the sake of consistency, we present all of the known results in the form of a join product instead of complete multipartite graphs. In particular, it is easy to check that for the graphs $K_2 + D_d$, the following is true.

Lemma 4.13. *For any positive integer d,*

$$cr(K_2 + D_d) = 0.$$

In the following, recall that $H(n) = (1/4) \left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor \left\lfloor \frac{n-2}{2} \right\rfloor \left\lfloor \frac{n-3}{2} \right\rfloor$ and $Z(n_1, n_2) = \left\lfloor \frac{n_1}{2} \right\rfloor \left\lfloor \frac{n_1-1}{2} \right\rfloor \left\lfloor \frac{n_2}{2} \right\rfloor \left\lfloor \frac{n_2-1}{2} \right\rfloor$.

For the graphs $K_3 + D_d$, the following holds by an argument similar to those used throughout Section 3.3 and although it was probably noticed much earlier, it is explicitly given in [117].

Lemma 4.14 (Qian and Huang, 2007 [117]). *For any positive integer d,*

$$cr(K_3 + D_d) = Z(d, 3).$$

The case $K_4 + D_d$ was determined by Ho in 2009 [81].

Theorem 4.15 (Ho, 2009 [81]). *For any positive integer d,*

$$cr(K_4 + D_d) = Z(d, 4) + d.$$

The final case for which the crossing numbers are known, $K_5 + D_d$ was determined by Ĺu and Huang in 2008 [105].

Theorem 4.16 (Ĺu and Huang, 2008 [105]). *For any positive integer d,*

$$cr(K_5 + D_d) = Z(d, 5) + 2d + \left\lfloor \frac{d}{2} \right\rfloor + 1.$$

Note that, if $d = 0$, then $K_n + D_d \cong K_n$ and, if $d = 1$, then $K_n + D_d \cong K_{n+1}$. Therefore, for the general case, we focus on $d \geq 2$. Additionally, note that, if $d = 2$, then $K_n + D_d$ is isomorphic to the complete graph on $n + 2$ vertices with one edge deleted. For this case, some crossing numbers for small instances are determined in [112] by adapting arguments originally used for complete graphs.

Theorem 4.17 (Ouyang et al., 2014 [112]). *For $n \leq 10$,*

$$cr(K_n + D_2) = \frac{1}{4} \left\lfloor \frac{n+4}{2} \right\rfloor \left\lfloor \frac{n+1}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor \left\lfloor \frac{n-2}{2} \right\rfloor.$$

For the general case, we derive a recursive lower bound by adapting a classical counting argument (for example, see [89]):

Lemma 4.18. *For $d \geq 2$,*

$$cr(K_n + D_d) \geq \frac{n\imath cr(K_{n-1} + D_d) + 2cr(K_{n+1})}{n-2}.$$

Proof. Suppose D is a good drawing of $K_n + D_d$. Let c be a crossing of

D between the edges (u, v) and (w, x). Then c falls into exactly one of the following three types:

1. $|\{u, v, w, x\} \cap V(D_d)| = 0$.
2. $|\{u, v, w, x\} \cap V(D_d)| = 1$.
3. $|\{u, v, w, x\} \cap V(D_d)| = 2$.

Let the number of crossings of types $1, 2, 3$ be denoted as c_1, c_2, c_3. Now consider the n subgraphs, along with their corresponding subdrawings in D, created when each vertex from $V(K_n)$ is deleted, one at a time. Each drawing of these subgraphs is a drawing of $K_{n-1} + D_d$, and hence, has at least $\mathrm{cr}(K_{n-1} + D_d)$ crossings. We can easily count the number of subgraphs, in which a particular crossing must be included. In particular, a crossing of type 1 is in exactly $(n - 4)$ of these subgraphs. A crossing of type 2 is in exactly $(n - 3)$ of these subgraphs. A crossing of type 3 is in exactly $(n - 2)$ of these subgraphs. Then,

$$(n-4)c_1 + (n-3)c_2 + (n-2)c_3 \geq ncr(K_{n-1} + D_d). \tag{4.2}$$

Suppose D is an optimal drawing of $K_n + D_d$. Then $c_1 + c_2 + c_3 = cr(K_n + D_d)$ and so,

$$(n-2)cr(K_n + D_d) - c_2 - 2c_1 \geq ncr(K_{n-1} + D_d). \tag{4.3}$$

Notice that c_1 corresponds to the number of crossings in a drawing of K_n. Next, by an averaging argument, there exists at least one vertex $y \in V(D_d)$ such that the number of crossings on edges incident to y, which are also type 2 crossings, is at least c_2/d. Adding these crossings to c_1, we obtain all crossings of a drawing of K_{n+1}, and so,

$$\frac{c_2}{d} + c_1 \geq cr(K_{n+1}). \tag{4.4}$$

and hence substituting the above into (4.3),

$$(n-2)cr(K_n+D_d)+\frac{2-d}{d}c_2-2cr(K_{n+1})\geq \\ ncr(K_{n-1}+D_d). \tag{4.5}$$

For $d=2$, the result follows immediately. For $d>2$ the second term on the left hand side is negative and so we may ignore it and the result follows. □

Upper bounds on the crossing number of $cr(K_n+D_d)$ can be shown by constructing a drawing procedure. In [120], Richter and Thomassen investigated so-called *cylinder drawings* of K_n. The cylinder drawings of K_n are a family of drawings of K_n where there are two concentric cycles with every vertex lying on one of the cycles and such that no crossings exist on either cycle. We observe that specific edges of the cylinder drawings have a large number of crossings on them and utilise this to derive an upper bound for $cr(K_n+D_d)$ by removing as many of these edges as possible. The derived bound obtains the exact value of $cr(K_n+D_d)$ for the known cases of $n=2,3,4,5$ as well as the cases described in Theorem 4.17, where $d=2$. When $d=0$, the derived bound coincides with $H(n)$, and when $d=1$, it coincides with $H(n+1)$ (recall that $K_n+D_1\cong K_{n+1}$). In the following, we will construct a particular drawing of K_n+D_d which we will denote a *spiral cylinder drawing*.

First, we begin with a well-known construction of a cylinder drawing of K_{n+d} and then delete the edges of a vertex induced subgraph of K_d to reduce it to a drawing of K_n+D_d. Construct a cylinder drawing of K_{n+d}, for $n+d\geq 6$, in the following way:

1. Place $\lceil (n+d)/2\rceil$ vertices on a small circle.
2. Draw straight lines between each of the $\lceil (n+d)/2\rceil$ vertices to form $\binom{\lceil (n+d)/2\rceil}{4}$ crossings.
3. Place $\lfloor (n+d)/2\rfloor$ vertices on a larger circle which encloses the small circle, in the configuration illustrated in Figure 4.11, depending on

whether $n+d$ is odd or even.

4. Connect the vertices of the larger circle, using the unbounded region, to form another $\binom{\lfloor (n+d)/2 \rfloor}{4}$ crossings.
5. Choose a vertex on the inner circle and connect it to each vertex on the outside circle, drawing the edges in a clockwise fashion.
6. Choose the next vertex anticlockwise on the inner circle and repeat the previous step.

Cylinder drawings for K_7 and K_8 are illustrated in Figure 4.11. Note that in those illustrations, the edges inside the inner cycle are omitted for clarity, as are the edges in the unbounded region. It is a simple exercise to count the crossings of this drawing for general n, and it coincides exactly with $H(n)$.

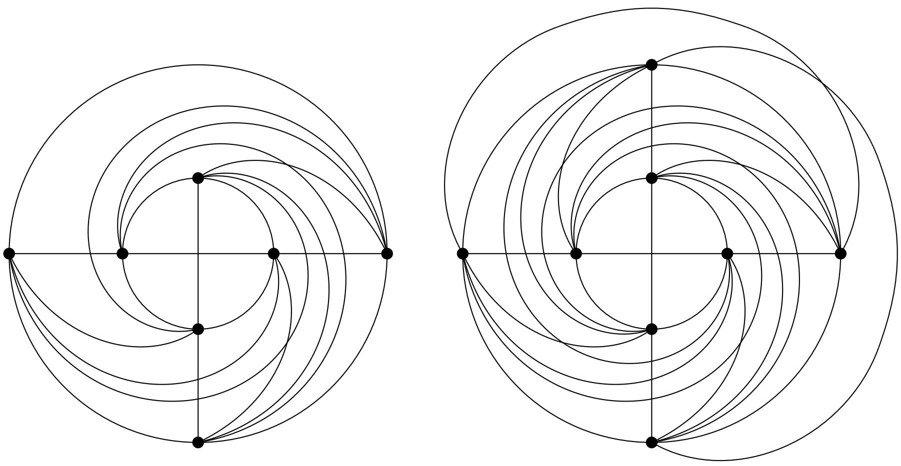


Figure 4.11: Cylinder drawings of K_7 on the left and K_8 on the right.

It is clear that, beginning with the graph K_{n+d}, the edges of a vertex induced subgraph formed by any set of d vertices can be deleted and the resulting graph is $K_n + D_d$. So, we will begin with the cylinder drawing of K_{n+d} described above, which has $H(n+d)$ crossings, and then count how many crossings are removed when edges of the drawing are deleted to obtain a drawing of $K_n + D_d$. Beginning with the vertex located at the top of the inner circle, label the vertices on the inner circle in a anticlockwise fashion as $a_1, a_2, \ldots, a_{\lceil (n+d)/2 \rceil}$. Then, beginning at the vertex immediately anticlockwise from the top vertex on the outside circle (if $n+d$ is even), or the vertex immediately anticlockwise from the top spot which is missing a vertex (if $n+d$ is odd), label the vertices on the outside circle in an anticlockwise fashion as $b_1, b_2, \ldots, b_{\lfloor (n+d)/2 \rfloor}$. Delete the edges of the vertex induced subgraph formed by the vertices $\{a_1, a_2, \ldots, a_{\lceil d/2 \rceil}\} \cup \{b_1, b_2, \ldots, b_{\lfloor d/2 \rfloor}\}$. We refer to the resulting drawing as a *spiral cylinder drawing* of $K_n + D_d$. Figure 4.12 displays example drawings of $K_3 + D_4$ and $K_4 + D_4$.

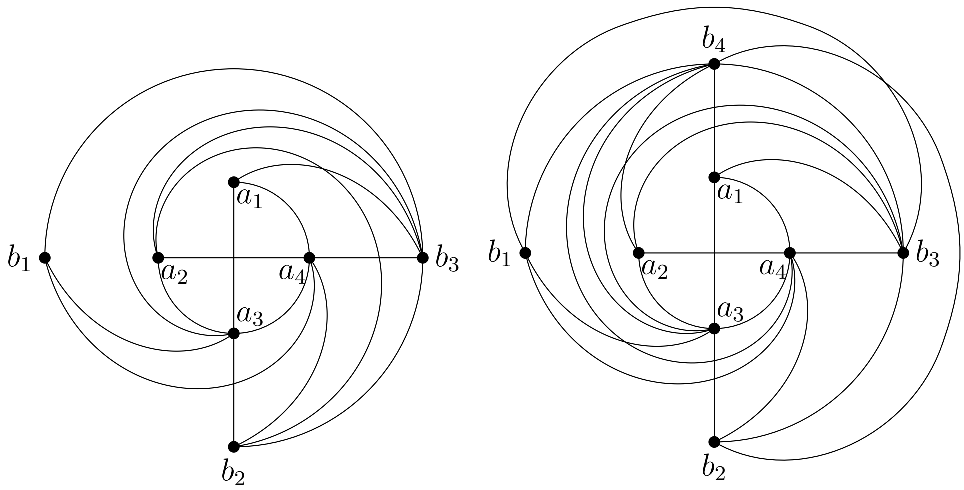


Figure 4.12: On the left, a spiral cylinder drawing of $K_3 + D_4$ and on the right, a spiral cylinder drawing of $K_4 + D_4$.

The proof of the upcoming theorem, although not particularly difficult, is rather tedious, and involves the consideration of many cases. For the sake of clarity, we omit the proof here; however, the full proof is included in Appendix B.

Theorem 4.19. *For $n + d \geq 6$, the number of crossings in a spiral cylinder drawing of $K_n + D_d$ is $Z(n,d) + H(n) + f(n,d)$ where,*

$$f(n,d) = \frac{1}{2}\left\lfloor \frac{nd}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor \left\lfloor \frac{n-2}{2} \right\rfloor$$

Theorem 4.19 leads directly to the following corollary.

Corollary 4.20. $cr(K_n + D_d) \leq Z(n,d) + H(n) + f(n,d)$ *where,*

$$f(n,d) = \frac{1}{2}\left\lfloor \frac{nd}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor \left\lfloor \frac{n-2}{2} \right\rfloor .$$

Having established a new upper bound for the crossings numbers of $K_n + D_d$, we were interested in determining how tight the upper bound is. We repeatedly submitted various instances of the graphs $K_n + D_d$ to Quickcross in an attempt to find any drawing with strictly fewer crossings than the upper bound in Theorem 4.19. In all tested cases, Quickcross was unable to do so. Furthermore, for all tested values of the parameters, Quickcross finds drawings with exactly the stated upper bound. In particular, Quickcross obtained drawings of $K_n + D_d$ meeting the bound precisely for all values of n and d such that $n + d \leq 50$. The results of this experiment lead us to propose the following conjecture.

Conjecture 4.21. $\mathrm{cr}(K_n + D_d) = Z(n, d) + H(n) + f(n, d)$ *where,*

$$f(n, d) = \frac{1}{2}\left\lfloor\frac{nd}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor\left\lfloor\frac{n-2}{2}\right\rfloor$$

Tables 4.13 and 4.14 display some formulas which result from Theorem 4.19 when n or d are fixed. For all of the known cases discussed at the beginning of this section, the formulas coincide with the exact crossing numbers.

n	**New upper bound**	**Known results**
2	0	0
3	$Z(3, d)$	$Z(3, d)$
4	$Z(4, d) + d$	$Z(4, d) + d$
5	$Z(5, d) + \left\lfloor\frac{5d+2}{2}\right\rfloor$	$Z(5, d) + \left\lfloor\frac{5d+2}{2}\right\rfloor$
6	$Z(6, d) + 6d + 3$	-
7	$Z(7, d) + 3\left\lfloor\frac{7d+6}{2}\right\rfloor$	-
8	$Z(8, d) + 18d + 18$	-
9	$Z(9, d) + 6\left\lfloor\frac{9d+12}{2}\right\rfloor$	-
10	$Z(10, d) + 40d + 60$	-
11	$Z(11, d) + 10\left\lfloor\frac{11d+20}{2}\right\rfloor$	-
12	$Z(12, d) + 75d + 150$	-

Table 4.13: Resulting formulas from the upper bound in Theorem 4.20 when n is fixed. Note that, for $n \leq 5$, the formulas coincide with the exact crossing numbers.

Lastly, we remark that the next case for which exact results are yet to

d	**New upper bound**
0	$H(n)$
1	$H(n+1)$
2	$\frac{1}{4}\left\lfloor\frac{n-1}{2}\right\rfloor\left\lfloor\frac{n-2}{2}\right\rfloor\left\lfloor\frac{n+4}{2}\right\rfloor\left\lfloor\frac{n+1}{2}\right\rfloor$
3	$Z(n,3)+\frac{1}{4}\left\lfloor\frac{n-1}{2}\right\rfloor\left\lfloor\frac{n-2}{2}\right\rfloor\left(\left\lfloor\frac{n+5}{2}\right\rfloor\left\lfloor\frac{n+4}{2}\right\rfloor-4\right)$
4	$Z(n,4)+\frac{1}{4}\left\lfloor\frac{n-1}{2}\right\rfloor\left\lfloor\frac{n-2}{2}\right\rfloor\left(\left\lfloor\frac{n+8}{2}\right\rfloor\left\lfloor\frac{n+5}{2}\right\rfloor-8\right)$
5	$Z(n,5)+\frac{1}{4}\left\lfloor\frac{n-1}{2}\right\rfloor\left\lfloor\frac{n-2}{2}\right\rfloor\left(\left\lfloor\frac{n+9}{2}\right\rfloor\left\lfloor\frac{n+8}{2}\right\rfloor-16\right)$
6	$Z(n,6)+\frac{1}{4}\left\lfloor\frac{n-1}{2}\right\rfloor\left\lfloor\frac{n-2}{2}\right\rfloor\left(\left\lfloor\frac{n+12}{2}\right\rfloor\left\lfloor\frac{n+9}{2}\right\rfloor-24\right)$
7	$Z(n,7)+\frac{1}{4}\left\lfloor\frac{n-1}{2}\right\rfloor\left\lfloor\frac{n-2}{2}\right\rfloor\left(\left\lfloor\frac{n+13}{2}\right\rfloor\left\lfloor\frac{n+12}{2}\right\rfloor-36\right)$
8	$Z(n,8)+\frac{1}{4}\left\lfloor\frac{n-1}{2}\right\rfloor\left\lfloor\frac{n-2}{2}\right\rfloor\left(\left\lfloor\frac{n+16}{2}\right\rfloor\left\lfloor\frac{n+13}{2}\right\rfloor-48\right)$
9	$Z(n,9)+\frac{1}{4}\left\lfloor\frac{n-1}{2}\right\rfloor\left\lfloor\frac{n-2}{2}\right\rfloor\left(\left\lfloor\frac{n+17}{2}\right\rfloor\left\lfloor\frac{n+16}{2}\right\rfloor-64\right)$
10	$Z(n,10)+\frac{1}{4}\left\lfloor\frac{n-1}{2}\right\rfloor\left\lfloor\frac{n-2}{2}\right\rfloor\left(\left\lfloor\frac{n+20}{2}\right\rfloor\left\lfloor\frac{n+17}{2}\right\rfloor-80\right)$
11	$Z(n,11)+\frac{1}{4}\left\lfloor\frac{n-1}{2}\right\rfloor\left\lfloor\frac{n-2}{2}\right\rfloor\left(\left\lfloor\frac{n+21}{2}\right\rfloor\left\lfloor\frac{n+20}{2}\right\rfloor-100\right)$
12	$Z(n,12)+\frac{1}{4}\left\lfloor\frac{n-1}{2}\right\rfloor\left\lfloor\frac{n-2}{2}\right\rfloor\left(\left\lfloor\frac{n+24}{2}\right\rfloor\left\lfloor\frac{n+21}{2}\right\rfloor-120\right)$

Table 4.14: Resulting formulas from the upper bound in Theorem 4.20 when d is fixed. Note that, for $d = 2$, the formula coincides with the exact crossing number, for the known cases of $n \leq 10$ (from Lemma 4.17).

be determined is $K_6 + D_d$. When n is even, the upper bound in Corollary 4.20 collapses nicely to $H(n) + Z(n,d) + d(H(n+1) - H(n))$. There is a neat interpretation of this formula. Consider one of the vertices from D_d. When the join product is performed, it along with K_n forms K_{n+1}. Hence, it adds some number of crossings to the drawing, and if K_n is drawn optimally, this number is at least $(cr(K_{n+1}) - cr(K_n))$. This argument can be made for each of the d vertices. Also, the addition of these edges for all d vertices collectively introduces $K_{n,d}$ as a subgraph, and so the number of crossings also increases by at least $cr(K_{n,d})$. Then, one way to complete a proof of $cr(K_n + D_d)$, would be to establish that if vertices of D_d can be joined to K_n with fewer than $(cr(K_{n+1}) - cr(K_n))$ crossings, then either the K_n or the $K_{n,d}$ are drawn sufficiently sub-optimally. For the case $K_6 + D_d$, the values in the above argument $cr(K_6)$, $cr(K_7)$ and $cr(K_{6,d})$ are all known and these graphs may be small enough to complete the argument via a brute force computation.

4.3 Generalised Petersen graphs

The generalised Petersen graph $GP(n,k)$ is defined as the graph on $2n$ vertices labelled as $\{u_0, \dots, u_{n-1}, v_0, \dots, v_{n-1}\}$ and with the edge set $\{(u_i, u_{i+1}),$ $(u_i, v_i), (v_i, v_{i+k}); i = 0, \dots, n-1\}$ where the subscripts are read modulo n. Figure 4.15 displays example drawings of $GP(6,2)$ and $GP(9,4)$. With this notation, $GP(n,k)$ is isomorphic to $GP(n, n-k)$ and so it is standard to only consider $k \leq n/2$. Then, in all cases except for $k = n/2$, generalised Petersen graphs are simple and 3-regular. It is worth noting that, in literature, it is common for generalised Petersen graphs to be denoted by $P(n,k)$. However, in order to avoid confusion with the notation for paths, which we consider extensively in this thesis, in what follows, we will use $GP(n,k)$ to denote a generalised Petersen graph.

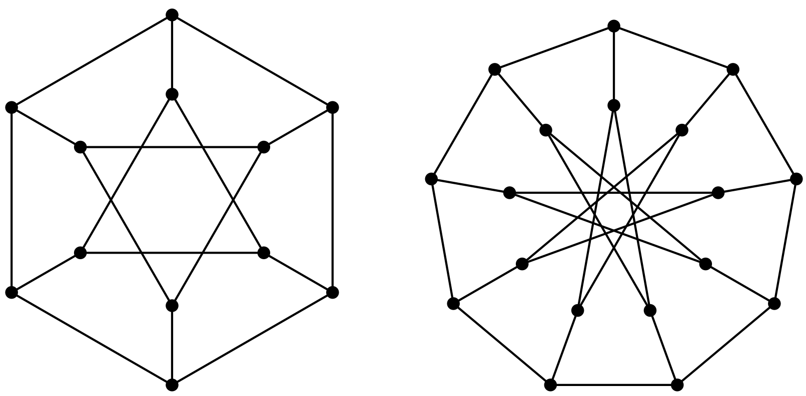


Figure 4.15: Drawings of $GP(6,2)$ on the left and $GP(9,4)$ on the right.

For some values of the parameters n and k, the crossing number of $GP(n,k)$ is known, and in fact, some of these graphs were considered in Section 2.5.2 when benchmarking the performance of Quickcross. It is also known that there are several different isomorphism classes for the generalised Petersen graphs and these are fully characterised in [133]. We will begin by considering the smallest unknown general case, $GP(n,4)$, for which only partial results are known. We will show that Quickcross appears to reliably find drawings for these graphs which coincide with the myriad known results and conjectures. Motivated by this, we use Quickcross to predict the crossing numbers for the next case, $GP(n, 5)$.

For $k = 1$, generalised Petersen graphs are always planar. The cases when $k = 2$ and $k = 3$ are now fully decided; the history of these cases is described in a recent survey [39].

In 1986, Fiorini [56] was the first to consider the case when $k = 4$. In particular, he considered $GP(4h, 4)$, and claimed the following result.

Theorem 4.22 (Fiorini, 1986 [56]). *For any integer $h \geq 4$,*

$$cr(GP(4h, 4)) = 2h.$$

This result has been accepted in literature. However, we note here that no proof was given in Fiorini's paper (which was primarily devoted to the case when $k = 3$), and to the best of our knowledge no subsequent paper has addressed this oversight. As part of a general investigation of the graphs $GP(n, 4)$, Chimani [30] has independently verified that the values of $cr(GP(4h, 4))$ provided by Theorem 4.22 are correct for all $h \leq 44$.

Lin et al. [144] provided a drawing procedure for each graph $GP(4h+2, 4)$, $h \geq 3$, which possesses $2h + 2$ crossings. This implies an upper bound and it is conjectured independently in [144] and [30] that equality holds.

Conjecture 4.23 (Lin et al., 2009 [144], Chimani, 2008 [30]). *For any integer $h \geq 3$,*

$$cr(GP(4h + 2, 4)) = 2h + 2.$$

Finally, Chimani [30] proposed conjectures for the two remaining cases $GP(4h + 1, 4)$ and $GP(4h + 3, 4)$, and verified the conjectures for many small and moderate values of h by utilising his integer programming exact methods.

Conjecture 4.24 (Chimani, 2008 [30])**.** *For any integer $h \geq 3$, and $k \in \{1, 3\}$,*

$$cr(GP(4h + k, 4)) = 2h + 4,$$

with the sole exceptions of $GP(13, 4)$ and $GP(17, 4)$ whose crossing numbers are 7 and 10 respectively.

For small values of n, some other special cases of $GP(n, 4)$ were first determined in [57, 42, 56, 70, 144, 107, 124] by utilising various theoretical methods and exact algorithms. These are summarised in Table 4.16. More recently, in [30], Chimani reconfirmed these values and extended the known values to considerably larger values of n.

n	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**
$cr(P(n, 4))$	1	3	4	5	4	7	8	10	8	10

Table 4.16: Some known crossing numbers of graphs $GP(n, 4)$ for small orders.

We ran Quickcross on the graphs $GP(n, 4)$, for $n = 8, 9, \ldots, 100$. The minimum obtained number of crossings for each n is displayed in Table 4.17, separated into the four cases listed above so to make the pattern obvious. For the graphs $GP(4h, 4)$, the crossing number is given in Theorem 4.22 and Quickcross obtains optimal drawings of $GP(4h, 4)$ for all $h < 20$. Beyond this point, Quickcross struggles to find optimal drawings and the discrepancy between the best found solution and $cr(GP(4h, 4))$ is between 0 and 2 for $20 \leq h \leq 25$. For the graphs $GP(4h+2, 4)$, the crossing number is not known, however the value in Conjecture 4.23 provides an upper bound. Quickcross obtains drawings of $GP(4h+2, 4)$ for which the number of crossings meets the upper bound for all $h < 18$. Similarly to the previous case, for $18 \leq h < 25$, the discrepancy between the best found solution and the upper bound is between 0 and 3. For the graphs $GP(4h+1, 4)$ and $GP(4h+3, 4)$, Quickcross

n	h 3	4	5	6	7	8	9	10	11	12	13	14	15	16
$4h$	4	8	10	12	14	16	18	20	22	24	26	28	30	32
$4h+1$	7	10	14	16	18	20	22	24	26	28	30	32	34	36
$4h+2$	8	10	12	14	16	18	20	22	24	26	28	30	32	34
$4h+3$	10	12	14	16	18	20	22	24	26	28	30	32	34	36

n	h 17	18	19	20	21	22	23	24	25
$4h$	34	36	38	41	42	46	47	49	52
$4h+1$	38	40	42	44	46	48	50	52	-
$4h+2$	36	39	40	43	47	48	51	53	-
$4h+3$	38	40	42	44	46	48	50	52	-

Table 4.17: Minimum found number of crossings by Quickcross for the graphs $GP(n,4)$.

was able to obtain a solution that agrees with Conjecture 4.24 in all cases for $h < 25$.

Although Quickcross was unable to obtain optimal solutions in all tested cases, for moderate sizes of n, it was successful in at least meeting the best known solutions. Buoyed by this outcome, we now investigate the next smallest unknown case, namely the graphs $GP(n,5)$, for which even conjectures of exact values for general n have not yet been published. Some values of $cr(GP(n,5))$ are known for small n. Table 4.18 displays these values and the results are due to [42, 56, 70, 144, 107, 124].

n	10	11	12	13	14	15	16
$cr(GP(n,5))$	1	3	8	9	6	5	8

Table 4.18: Known crossing numbers of graphs $GP(n,5)$ for small orders.

We note here that in [42], it was claimed that $cr(GP(17,5)) = 14$, however, in our experimentation, Quickcross finds a drawing of $GP(17,5)$ with 13 crossings. One such drawing of $GP(17,5)$ is displayed in Figure 4.19, and so we conclude that there was an error in the arguments of [42].

In Table 4.20, we display the minimum obtained number of crossings from

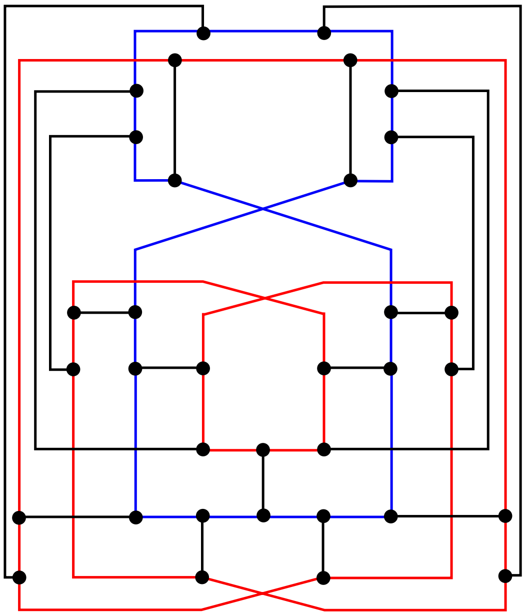


Figure 4.19: $GP(17,5)$ drawn with 13 crossings. The blue edges form the cycle $u_0, u_1, \ldots, u_{16}, u_0$ and the red edges form the cycle with edges $\{(v_i, v_{i+5})\}$ for $i = 0, \ldots, d-1$.

	h													
n	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
$5h$	1	5	11	18	24	28	32	36	40	45	49	53	57	61
$5h+1$	3	8	14	22	26	30	34	38	43	46	50	54	59	63
$5h+2$	8	13	19	24	28	32	36	40	44	48	53	57	61	65
$5h+3$	9	14	19	24	28	32	36	40	44	49	53	57	60	65
$5h+4$	6	12	20	24	28	32	36	40	45	49	52	56	61	65

Table 4.20: Minimum found number of crossings by Quickcross for the graphs $GP(n,5)$.

Quickcross for values of $n \leq 79$. After observing the clear pattern in Table 4.20, the following conjecture appears appropriate.

Conjecture 4.25. *For any integer $h \geq 5$,*

1. $cr(GP(5h,5)) = 4h$,
2. $cr(GP(5h+1,5)) = 4h+2$,
3. $cr(GP(5h+2,5)) = 4h+4$,
4. $cr(GP(5h+3,5)) = 4h+4$,
5. $cr(GP(5h+4,5)) = 4h+4$,

With the exception of $GP(25,5)$, for which $cr(GP(25,5)) = 18$.

The first step in seeking to confirm Conjecture 4.25 is to provide upper bounds meeting the conjectured values. As usual, this can be achieved by providing appropriate drawing procedures. However, it is not trivial to discover such drawing procedures in these cases. In order to aid us in doing so, we took advantage of the output from Quickcross.

Our hope was to discover a drawing procedure which generalises for any value of n. To start with, we considered small values of n, and attempted to produce a nicely structured drawing with the conjectured number of crossings. In an ideal situation, such a drawing can be generalised to larger values of n. However, there may be many possible drawings with the conjectured number of crossings that are not easily generalisable to larger values of n. Identifying such a drawing that is amenable to such generalisation is a challenge in its own right.

Each time Quickcross is run, we obtain such a drawing (discarding the runs where the conjectured number of crossings is not reached). The drawing is not provided graphically; it is captured inside of Quickcross' data structures, as discussed in Chapter 2. Of course, any planar graph drawing procedure (applied to the planarisation of the drawing) can be used to produce a graphical drawing, however such procedures do not usually provide an easily interpretable drawing. Hence, it is typically necessary to manually play with the drawing to determine if it is useful.

Obviously, there are countless possible drawings, and most of them are unlikely to lead to an easy generalisation. It is undesirable to manually consider each of them individually. To address this, we conducted the following experiment. We repeatedly ran Quickcross, and recorded the various produced drawings. Then, we analysed the data structures of these drawings to identify only those with desirable properties for further analysis. However, it was not clear in advance what properties are desirable in a drawing. We experimented with various properties until we discovered one that led to nicely structured drawings, which we describe now.

Recall that a generalised Petersen graph has vertices labelled $\{u_0, u_1, \ldots, u_{n-1}, v_0, \ldots, v_{n-1}\}$. Define $U := \{u_0, \ldots, u_{n-1}\}$ and $V := \{v_0, \ldots, v_{n-1}\}$. Then its edges can be partitioned into three disjoint sets; edges whose vertices are both contained in U, edges whose vertices are both containted in V, and then the remaining edges with one vertex from U and one from V. Respectively, we will refer to these three sets of edges as *blue* edges, *red* edges, and *black* edges. Then, the property that led to nicely structured drawings is as follows. We requested that none of the black edges cross one another, and looked for a minimal number of crossings between red and blue edges. Drawings that obeyed this rule became increasingly rare as we increased n. However, after running Quickcross millions of times on many instances, we were able to find some good candidate drawings, and consider them more closely.

Following this experiment, we are now able to give drawing procedures for $GP(n, 5)$ which meet our conjectured values. Specifically, we now show that the following upper bounds on $cr(GP(n, 5))$ hold.

Theorem 4.26. *For any integer $h \geq 5$,*

1. $cr(GP(5h, 5)) \leq 4h$,
2. $cr(GP(5h + 1, 5)) \leq 4h + 2$,
3. $cr(GP(5h + 2, 5)) \leq 4h + 4$,
4. $cr(GP(5h + 3, 5)) \leq 4h + 4$,
5. $cr(GP(5h + 4, 5)) \leq 4h + 4$,

With the exception of $GP(25, 5)$, for which $cr(GP(25, 5)) \leq 18$.

Proof. We present five drawing procedures for the five cases above. Each procedure contains two parts; one part which remains invariant for different values of h, and one part in which the extra vertices are added when considering different values of h. We shall call the invariant part the *base* of the construction. In each case, the number of crossings can be easily counted by first counting the crossings in the base. In each case, the base corresponds to a drawing of the graph for a particular value of h. Then, each time h is increased by one, we introduce ten new vertices. Five of the new vertices are added to the cycle of blue edges, and each of them are attached to a corresponding new vertex on the cycle(s) of red edges. This is then simply repeated for each subsequent increase in h. In each case, it will be seen that adding in these ten new vertices and their edges introduces exactly four crossings into the drawing (shown in the example Figures).

For the graphs $GP(5h, 5)$, Figure 4.21 displays a drawing of $GP(25, 5)$ with 18 crossings. Then, for general case, the base of the construction comes from $GP(30, 5)$, that is, $h = 6$. Figure 4.22 displays the base of the construction which contains $24 = 4h$ crossings. The right hand side of Figure 4.23 shows the additions needed to produce drawings of the next case $GP(35, 5) = GP(5(h + 1), 5)$, with exactly $4(h + 1)$ crossings.

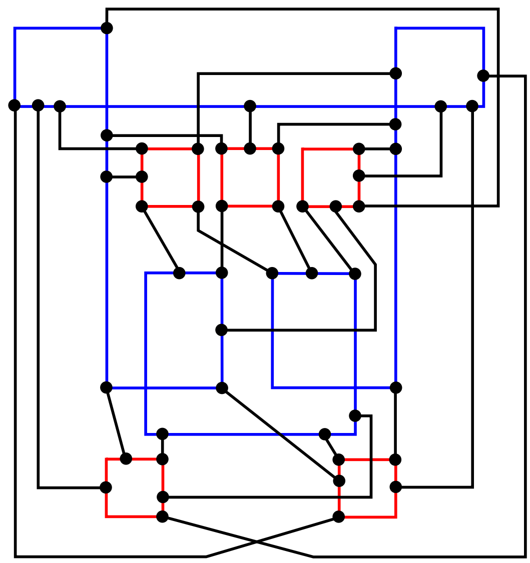


Figure 4.21: $GP(25, 5)$ drawn with 18 crossings.

For the graphs $GP(5h + 1, 5)$, the base of the construction comes from $GP(26, 5)$, that is, $h = 5$. Figure 4.24 displays the base of the construction which contains $22 = 4h + 2$ crossings. The right hand side of Figure 4.25 shows the additions needed to produce drawings of the next case $GP(31, 5) =$

$GP(5(h+1)+1,5)$, with exactly $26 = 4(h+1)+2$ crossings.

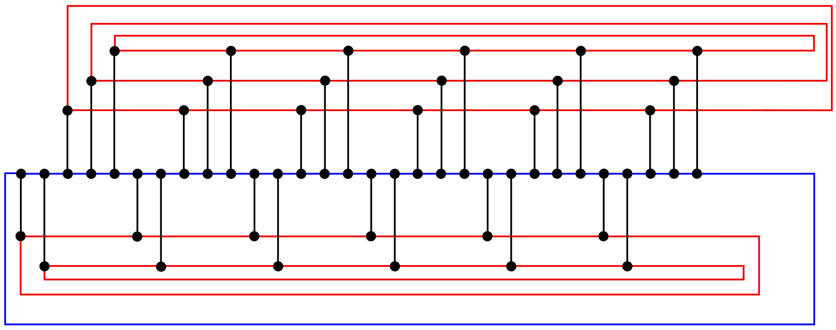


Figure 4.22: $GP(30,5)$ drawn with 24 crossings.

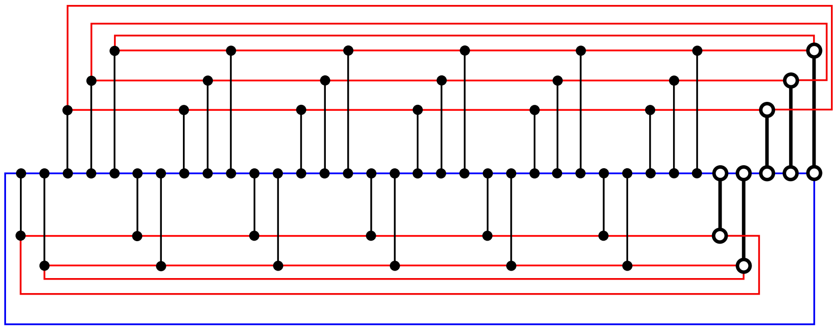


Figure 4.23: $GP(35,5)$ drawn with 28 crossings.

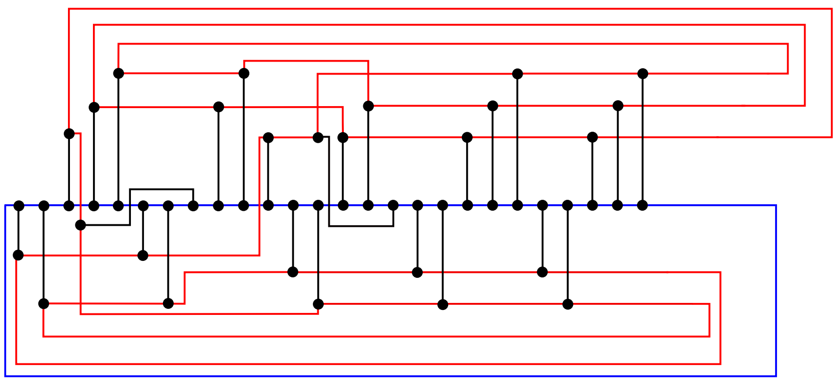


Figure 4.24: $GP(26,5)$ drawn with 22 crossings.

For the graphs $GP(5h+2,5)$, the base of the construction comes from $GP(27,5)$, that is, $h=5$. Figure 4.26 displays the base of the construction which contains $24 = 4h+4$ crossings. The right hand side of Figure 4.27 shows the additions needed to produce drawings of the next case $GP(32,5) = GP(5(h+1)+2,5)$, with exactly $28 = 4(h+1)+4$ crossings.

For the graphs $GP(5h+3,5)$, the base of the construction comes from $GP(28,5)$, that is, $h=5$. Figure 4.28 displays the base of the construction

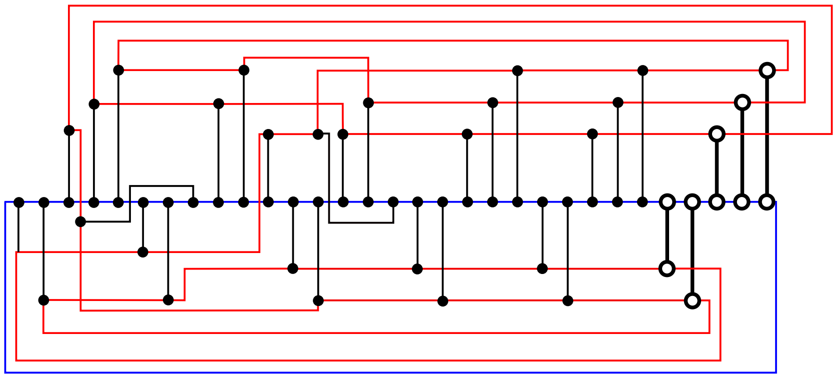


Figure 4.25: $GP(31,5)$ drawn with 26 crossings.

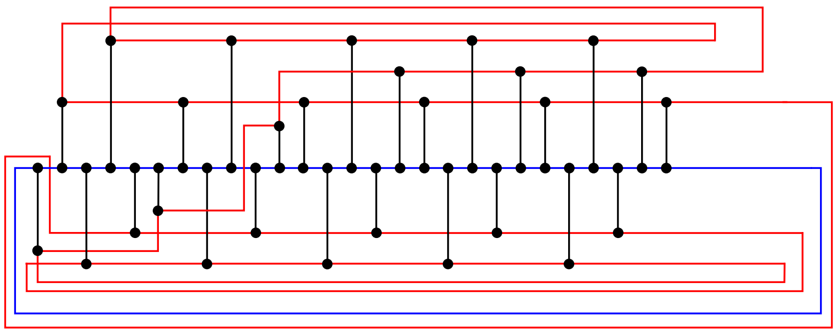


Figure 4.26: $GP(27,5)$ drawn with 24 crossings.

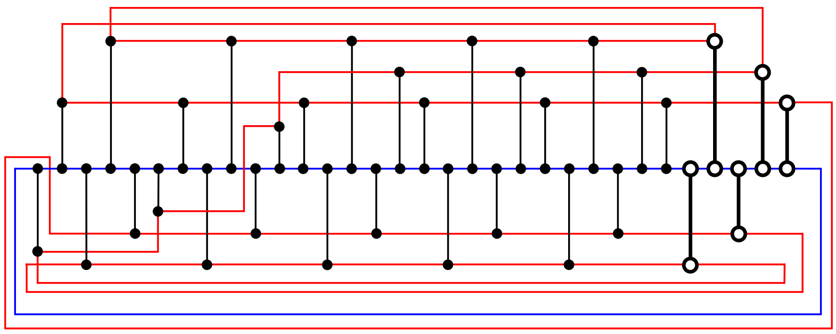


Figure 4.27: $GP(32,5)$ drawn with 28 crossings.

which contains $24 = 4h+4$ crossings. The right hand side of Figure 4.29 shows the additions needed to produce drawings of the next case $GP(33,5) = GP(5(h+1)+3,5)$, with exactly $28 = 4(h+1)+4$ crossings.

For the graphs $GP(5h+4,5)$, in this final case, the base of the construction comes from $GP(24,5)$, that is, $h=4$. Figure 4.30 displays the base of the construction which contains $20 = 4h+4$ crossings. The right hand side of Figure 4.31 shows the additions needed to produce drawings of the next case $GP(29,5) = GP(5(h+1)+4,5)$, with exactly $4(h+1)+4$ crossings. □

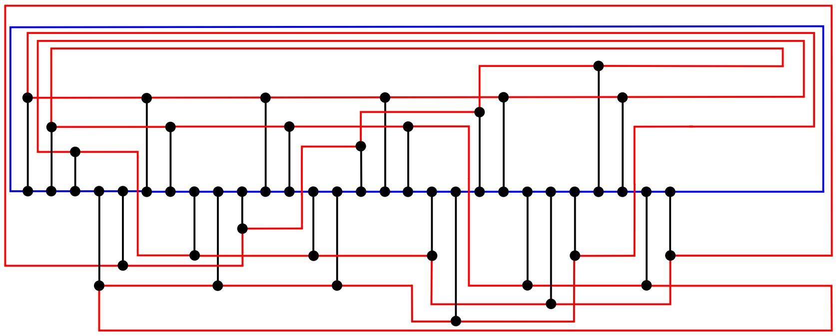


Figure 4.28: $GP(28,5)$ drawn with 24 crossings.

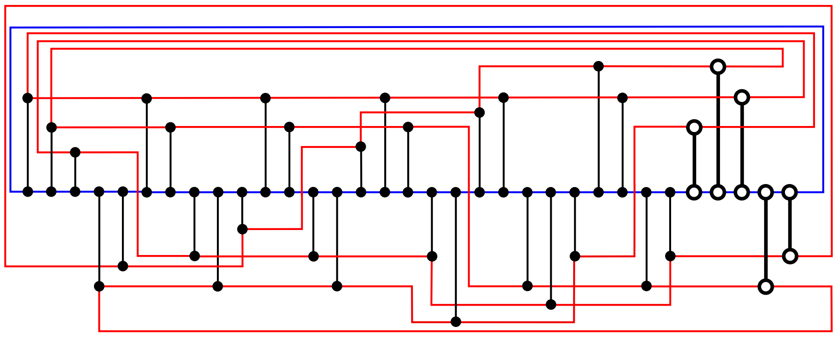


Figure 4.29: $GP(33,5)$ drawn with 28 crossings.

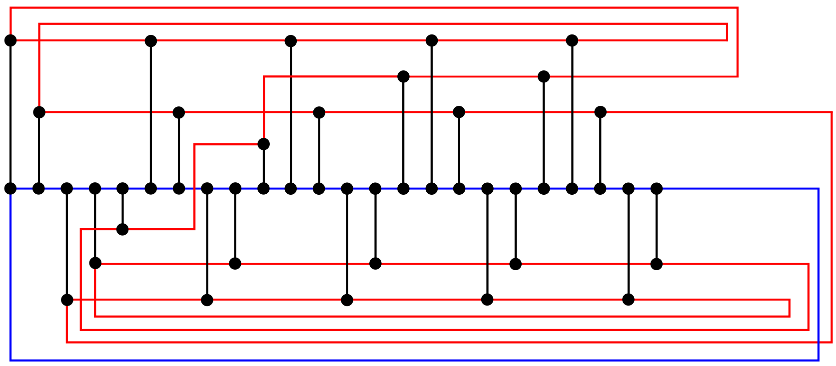


Figure 4.30: $GP(24,5)$ drawn with 20 crossings.

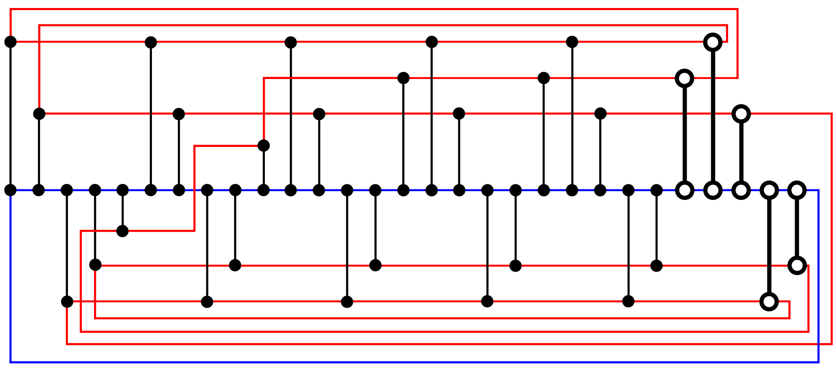


Figure 4.31: $GP(29,5)$ drawn with 24 crossings.

Recall that for $n \leq 16$, the values of $cr(GP(n, 5))$ are known. For $n \geq 25$, we have now established an upper bound for $cr(GP(n, 5))$. For the in-between cases where $17 \leq n \leq 24$, the values given in Table 4.20 provide upper bounds on $cr(GP(n, 5))$. Although for these cases, drawings which meet the upper bounds do not in general have the same structure as those given by the drawing procedures in Theorem 4.26, they can be easily retrieved from the output of Quickcross.

4.4 n-dimensional hypercube graphs

The n-dimensional hypercube graph, commonly referred to as the n-cube, is the graph whose vertices and edges are those of the n-dimensional hypercube. Let Q_n denote the n-cube, then $|V(Q_n)| = 2^n$ and $|E(Q_n)| = 2^{n-1}n$. A drawing of Q_4 is displayed in Figure 4.32. The crossing number of the n-cube is the subject of long standing conjectures. In 1970, Eggleton and Guy [51] claimed to have determined an upper bound on the crossing number of Q_n. However their proof was subsequently shown to contain an error and the proposed upper bound remained a conjecture for nearly 40 years. Despite this, it was widely believed to be accurate with Erdős and Guy even conjecturing that equality would hold [53]. Finally, in 2008, a drawing procedure was found by Faria et al. [55] which confirmed the conjectured upper bound.

Theorem 4.27 (Faria et al., 2008 [55])**.**

$$cr(Q_n) \leq \frac{5}{32}4^n - \left\lfloor \frac{n^2+1}{2} \right\rfloor 2^{n-2}.$$

Recently, in a yet to be published manuscript [145] by Yang et al., it is claimed that the methods of Faria et al. also contained a hiatus which is corrected by Yang et al. However, they also obtain new superior upper bounds. They claim to improve upon the bound in Theorem 4.27 by showing

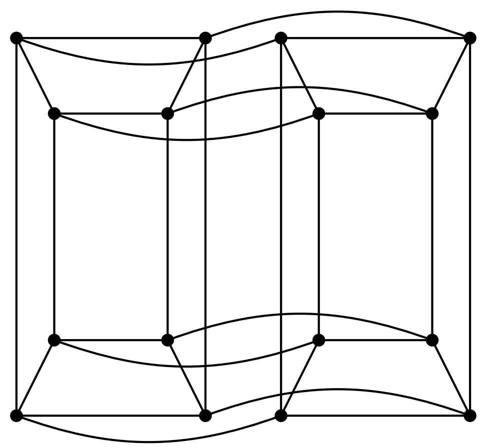


Figure 4.32: The 4-dimensional hypercube graph.

the following.

Theorem 4.28 (Yang et al., 2017 [145])**.**

$$cr(Q_n) \leq \begin{cases} \frac{139}{896}4^n - \left\lfloor \frac{n^2+1}{2} \right\rfloor 2^{n-2} + \frac{4}{7}2^{3\left\lfloor \frac{n}{2} \right\rfloor - n}, & \textit{if } 5 \leq n \leq 10 \\ \frac{26695}{172032}4^n - \left\lfloor \frac{n^2+1}{2} \right\rfloor 2^{n-2} - \frac{n^2+2}{2}2^{n-2} + \frac{4}{7}2^{3\left\lfloor \frac{n}{2} \right\rfloor - n}, & \textit{if } n \geq 11. \end{cases}$$

Theorem 4.28 implies that the prediction of Erdős and Guy that equality holds in Theorem 4.27 was incorrect. Theorem 4.28 also indicates that it may yet be possible to find even better bounds for large values of n. Some lower bounds for $cr(Q_n)$ are also known, and the best of these is due to Sýkora and Vrt'o [135], who showed the following.

Theorem 4.29 (Sýkora and Vrt'o, 1993 [135])**.**

$$cr(Q_n) \geq \frac{4^n}{20} - (n^2+1)2^{n-1}.$$

Exact crossing numbers for the n-cube are only known for the first few cases. Trivially, $cr(Q_3) = 0$ and it was shown in [43] that $cr(Q_4) = 8$. The crossing number of Q_n for $n \geq 5$ has not yet been established.

We ran Quickcross on the graphs Q_n for $n \leq 10$, and compared the best obtained solutions to the upper bounds. These results are displayed in Table 4.33. Note that the number of vertices increases by a factor of two with each n and for the cases of $|V(Q_9)| = 512$ and $|V(Q_{10})| = 1024$, Quickcross

n	**Quickcross**	**Upper bound (Faria et al.)**	**Upper bound (Yang et al.)**
3	0	0	0
4	8	8	8
5	56	56	56
6	352	352	352
7	1758	1760	1744
8	8168	8192	8128
9	36386	35712	35424
10	157647	151040	149888
11	-	624128	619456

Table 4.33: The minimum achieved crossings by Quickcross for the n-cubes. Values from the previously best upper bound in Theorem 4.27 and the recent superior upper bound in Theorem 4.28 are also displayed.

runs in a matter of minutes, but for $|V(Q_{11})| = 2048$ an initial embedding typically has around 750,000 crossings and the vectors corresponding to combinatorial embeddings become too large to store and manipulate efficiently in our implementation. For $n \leq 6$ we obtained the upper bounds of Faria et al. and Yang et al., which agree for $n \leq 6$. Interestingly, for $7 \leq n \leq 8$ we found solutions which contained fewer crossings than the upper bound of Faria et al. These solutions, discovered independently from [145], confirm that equality does not hold in Theorem 4.27. Despite significant additional effort, we were not able to obtain a solution to Q_6 with less than the upper bound of 352 crossings and hence we predict that $n = 7$ is the smallest n for which Theorem 4.27 does not hold with equality. More precisely, we predict that indeed $cr(Q_5) = 56$ and $cr(Q_6) = 352$. Note that this also agrees with Theorem 4.28.

4.5 Sheehan's maximal uniquely Hamiltonian graphs

In [130], Sheehan showed that for a graph on $n \geq 4$ vertices, the maximal number of edges such that there exists precisely one Hamiltonian cycle, is $m = \lfloor n^2/4 \rfloor + 1$. Sheehan claimed that there was a unique such graph for each $n \geq 4$ and gave the construction; however, it was later discovered by Barefoot and Entringer [15] that the graphs are not unique. Nonetheless Sheehan's construction is fascinating in its own right and we will refer to the set of graphs obtained by his construction as the Sheehan graphs and denote them as $Sh(n)$. The Sheehan graph on n vertices is constructed in the following way. Begin with a cycle on n vertices with the vertices labelled $v_1, v_2, \ldots, v_n$ in a cyclic fashion. Then, for every even labelled vertex v_d, add edges going to each of the vertices $\{v_{d+2}, v_{d+3}, \ldots, v_n\}$. Figure 4.34 displays an example drawing of $Sh(16)$, with the unique Hamiltonian cycle occurring on the boundary.

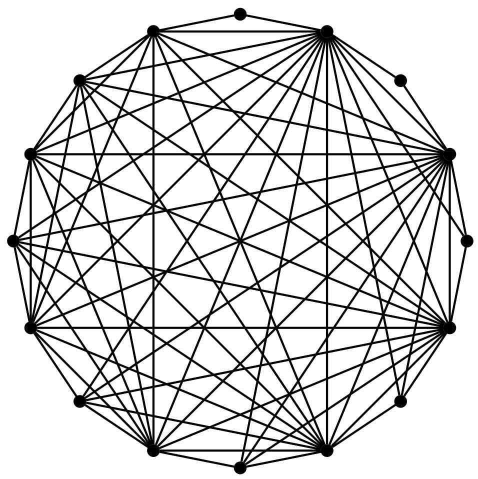


Figure 4.34: Sheehan graph on 16 vertices.

Although we are not concerned with the Hamiltonian property of Sheehan graphs, they are a family of graphs which are dense and which also contain a large amount of variation in their degree sequence. Indeed, a Sheehan graph on n vertices has the ordered degree sequence $\{2, 2, 3, \ldots, \lceil \frac{n+1}{2} \rceil, \lceil \frac{n+1}{2} \rceil$ $, \lceil \frac{n+1}{2} \rceil + 1, \ldots, n-2, n-1\}$. In this sense, they are the most irregular graphs upon which we report results in this thesis. The density of Sheehan graphs leads to difficulties when attempting to find the unique Hamiltonian cycle, even for state-of-the-art Hamiltonian Cycle Problem solvers [14]. Motivated by this, we were interested to investigate whether the Sheehan graphs also provided difficult instances for crossing minimisation. The crossing number of the Sheehan graphs has not previously been investigated and so we only make some preliminary observations here. First, the following is a simple observation, but we include it here as an initial result for these graphs.

Lemma 4.30.

$$cr(Sh(n)) = \Omega(n^4)$$

Proof. For each odd labelled vertex v_k, delete all edges which are incident to v_k, except (v_k, v_{k-1}) and (v_k, v_{k+1}). The remaining graph is a subdivision of $K_{\lfloor \frac{n}{2} \rfloor}$, hence this subgraph contributes at least $cr(K_{\lfloor \frac{n}{2} \rfloor})$ crossings to any drawing of $Sh(n)$. A counting argument, similar to that which is used in Lemma 4.2 of Section 4.18, can be used to show that $cr(K_{\lfloor \frac{n}{2} \rfloor}) = \Omega(n^4)$. □

After testing the Sheehan graphs with Quickcross, we observe that they provide a family of very difficult instances. Even for moderate values of n, after running $Sh(n)$ many thousands of times, Quickcross occasionally still improves upon its best found solution. We suspect that this difficulty is also present for other crossing minimisation heuristics and we attempted to investigate this by also running $Sh(n)$ with OGDF's implementation of the planarisation method. As discussed in Section 2.5, some schemes of OGDF lose tractability on dense graphs, and so we used the schemes $(fixed, all)$, $(variable, all)$ and $(multi, all)$. We observe that OGDF also had difficulty finding solutions with few crossings and these results are displayed in Table 4.35. We note that, the results between Quickcross and OGDF should not be compared as the graphs were run with a vastly different number of random permutations over an extended period of time.

Given the difficulty we encountered in running these graphs to optimality, we cannot be confident that the numbers contained in Table 4.35 coincide with the crossing numbers, particularly for the larger instances considered. This makes it difficult to predict a formula for the crossing number of $Sh(n)$. Nonetheless, for $n \leq 20$ both Quickcross and OGDF agree with one another, and for those cases we observe that the best found solutions closely, but not exactly, obey the following formula.

$$f(n) := \left\lfloor \frac{1}{320} n(n-2)(n-4)(n-6) \right\rfloor .$$

Given that, for $n \leq 20$, the differences between the best found solutions and $f(n)$ all fall within ± 1, we predict that the crossing number of Sheehan graphs lies within the range $f(n) \pm O(1)$. However, until we are able to more effectively handle the larger instances, there is no way of verifying whether this continues to be a reasonable assumption as n increases.

n	Quickcross	OGDF	$f(n)$	n	Quickcross	OGDF	$f(n)$
7	0	0	0	19	196	196	196
8	1	1	1	20	253	253	252
9	3	3	2	21	321	320	317
10	6	6	6	22	398	399	396
11	10	10	10	23	490	492	487
12	18	18	18	24	597	600	594
13	28	28	28	25	722	725	716
14	42	42	42	26	861	870	858
15	59	59	60	27	1022	1033	1018
16	84	84	84	28	1205	1217	1201
17	114	114	113	29	1411	1424	1406
18	152	152	151	30	1642	1665	1638

Table 4.35: The minimum achieved crossings by Quickcross and OGDF for the Sheehan graphs on n vertices. The values of the formula $f(n)$ are also displayed.

Chapter 5

Conclusions and future work

We conclude this thesis with a summary of the results obtained, and a discussion about the potential future research resulting from the work within each chapter.

In Chapter 2, we designed and implemented a crossing minimisation heuristic which uses solutions to the star insertion problem as its main workhorse. Significant effort was spent making our implementation efficient for practical use and we named the implementation Quickcross. We discussed the various features included within Quickcross and also some of the important algorithmic design decisions that were made. Then, we experimentally compared the different schemes of Quickcross by running it on some well known sets of graphs that have previously been used to compare crossing minimisation heuristics. In addition to these sets, we were also interested in the performance on dense graphs and chose some complete and complete bipartite graphs to run and make comparisons on. We identified several consistently strong performing scheme combinations for Quickcross and also observed others which struggled, specifically on the dense graphs.

Then, in Chapter 3, we used Quickcross to aid in developing new results. First, we extended some known results about the smallest cubic graphs with crossing number at least k. Along the way, we also determined some new values for the largest crossing number of any cubic graph on n vertices. Next, we studied the crossing number of the Cartesian product of a sunlet graph and a star. We found that the usual induction method for proving the crossing numbers of these types of families extended nicely to these graphs and we proved the crossing number of the first few cases of this family. We also made a conjecture about the crossing number for the general case and showed an upper bound that meets the conjectured value. We then investigated the crossing numbers of families of graphs resulting from a graph product of a fixed small graph with arbitrarily large cycles, paths, stars and discrete graphs. We were able to obtain the crossing numbers for 29 new such families of graphs. To investigate the many remaining gaps in known results of this type, we used Quickcross to predict what the crossing number of the unknown cases might be. As evidence to the strength of our predictions, we demonstrated that Quickcross also accurately predicted the crossing number for all of the known cases.

In Chapter 4, we considered graph families which are difficult to attack heuristically, either due to size, density or structure. First, we discussed the crossing number of the Cartesian product of a fixed graph with an arbitrarily large cycle. We observed that all known results about such graphs obeyed a simple formula and we conjectured this to be the case for the Cartesian product of any fixed graph with an appropriately large cycle, and provided substantial empirical evidence that this was the case. Next, we studied the crossing number of the join product of a complete graph and an empty graph. We determined lower bounds with a counting argument and then determined upper bounds with a drawing procedure. For all cases where the crossing number is known, our upper bounds coincide with the crossing number. We then studied the crossing number of the generalised Petersen graphs. With the aid of Quickcross, we were able to predict what the crossing number of the family $GP(n, 5)$ could be. We also used Quickcross to search for easily generalisable drawings for this case, and hence determined upper bounds for the crossing number coinciding with our predictions. Next, we studied the crossing number of the n-cube. Quickcross obtained a drawing of the 7-cube and 8-cube which possesses strictly fewer crossings than a long assumed tight upper bound. Although we were not the first researchers to notice this, our observation, independently from the recent unpublished work of Yang et al. [145], confirmed that the aforementioned upper bound is not tight. Finally, we made some preliminary investigations into crossing number of the Sheehan graphs. We observed that Sheehan graphs provided instances of dense graphs for which crossing minimisation heuristics have significant difficulties finding near optimal drawings. We also demonstrated a function which gives values extremely close to the best found results from Quickcross.

5.1 Future work arising from Chapter 2

In Chapter 2, we detailed the implementation of Quickcross, which we have heavily optimised. As such, there are no real directions of future work in terms of improving Quickcross itself. However, there are several interesting additions to Quickcross which could be futher considered.

In Section 2.6, we investigated a preliminary implementation of an analogue to the incremental post-processing strategy which was first introduced in [31]. The results from our implementation were promising, even with our rudimentary design and expectedly higher runtime. We expect that an efficient implementation of this method will produce one of the best practical crossing minimisation heuristics for relatively sparse graphs. Creating such an efficient implementation would be beneficial to the community of researchers investigating crossing minimisation.

In a recent line of research, Leaños and Salazar [100] along with Bokal et al. [22], studied the additivity of crossing numbers over minimal edge cut sets in graphs. Specifically, they determined conditions under which the crossing number of a graph is the sum of crossing numbers of its augmented components when minimal edge cut sets have been deleted. These results could be applied to provide a more sophisticated preprocessing scheme for Quickcross, and would provide significant benefits when running Quickcross on specific kinds of large graphs. Such a preprocessing scheme would require the identification of minimal edge cut sets and these can be found in polynomial time by the likes of the Edmonds-Karp algorithm for maximum flow.

A common disadvantage of all current crossing minimisation heuristics is their efficiency on dense graphs. It would be interesting to investigate methods which are specifically designed to aid crossing minimisation heuristics in solving dense graphs. Such a method may also lead towards a better understanding of some of the most important open problems in this area, which concern dense graphs.

5.2 Future work arising from Chapter 3

In Section 3.1, we extended the known results about minimal cubic graphs with crossing number k. At first, it seems unlikely that our same approach can be used to further extend these results, simply due to the enormous number of graphs needing considered to decide the next open case. However, if it was possible to reduce the number graphs which need to be considered by a theoretical argument, then new results may follow. For example, we were able to eliminate the graphs of girth three, but if graphs of even larger girth could also be eliminated, we expect that this would lead to further results.

In Section 3.3, we investigated the full catalogue of known results about crossing numbers of products of certain fixed small graphs with paths, cycles, stars and discrete graphs. Perhaps the next natural step in this direction is to strive for a complete list of such results for all fixed graphs on five and six vertices. Often these results have reasonably simple, but ad hoc proofs, and we expect that many cases remain undecided simply because they have not yet been fully investigated. Also, until the very recent survey [39] detailed the full list of known results, it was common for researchers to waste time reproving existing results. After taking into account the new results discovered in this thesis, there are 607 families (including both Cartesian and join products) whose crossing number remains to be determined. To this end, we now rank each of these families in order of importance by assigning a number p to each family. The number p is based on the following: if the crossing number of that family is proved, then p is the number of additional families with unknown crossing number whose proof follows as a corollary. These additional results can be obtained by arguments analogous to those used throughout Section 3.3. In Table 5.1, we display those families with unknown crossing number for the largest values of p.

5.3 Future work arising from Chapter 4

In Section 4.2, we determined lower and upper bounds for graphs resulting from the join product of a complete graph and a discrete graph. Of course, this family contains both K_n and $K_{n,d}$ as a subgraph and so one could not hope to prove any substantial exact results about this family without first knowing the crossing number of K_n and $K_{n,d}$. Nevertheless, the crossing number K_6 and $K_{6,d}$ are both known and yet the crossing number of $K_6 + D_d$ is yet to be determined. Although we were unsuccessful in proving this crossing number, we remain confident that it does in fact coincide with our upper bound. This result would also fill one of the remaining gaps in Table 3.34 of Section 3.3 for the graph $G_6^{156} + D_n$.

In Section 4.3, we determined upper bounds on the crossing number of

Table 5.1: For the families in Section 3.3 resulting from graph products and with unknown crossing number, we rank their importance by the number p of additional results which would immediately follow from a proof. The indices for those additional families are also provided.

$G_i^6 \Box P_n$

i	p	Additional results (G_i^6)					
102	6	107	114	122	123	124	127
58	3	115	116	133			
105	3	126	129	143			

$G_i^6 \Box C_n$

i	p	Additional results (G_i^6)				
93	5	103	121	128	137	140
31	4	48	72	73	79	
99	4	107	115	118	127	
48	3	72	73	79		
81	3	106	107	127		
101	3	112	122	134		
121	3	128	137	140		

$G_i^6 \Box S_n$

i	p	Additional results (G_i^6)			
51	4	65	70	89	90
105	4	126	129	141	143
44	3	66	74	83	
81	3	106	107	127	
88	3	115	116	133	
99	3	115	118	133	

$G_i^6 + D_n$

i	p	Additional results (G_i^6)								
7	9	10	14	18	19	21	28	33	43	46
14	7	19	21	26	28	41	43	46		
13	6	24	27	36	47	54	64			
37	6	56	62	76	89	90	98			

$G_i^6 + P_n$

i	p	Additional results (G_i^6)								
7	9	10	14	18	19	21	28	33	43	46
13	6	24	27	36	47	54	64			
14	6	19	21	26	28	43	46			

$G_i^6 + C_n$

i	p	Additional results (G_i^6)						
13	7	24	27	36	45	47	54	64
11	6	32	42	44	45	52	63	
21	6	35	36	47	63	64	66	
22	6	35	42	44	45	63	66	
37	6	56	62	76	89	90	98	

the graphs $GP(n, 5)$. The drawing procedures that were developed have reasonably simple generalisations to the graphs $GP(n, k)$ for $k > 5$. Interestingly, we suspect that these generalisations may provide an upper bound for $cr(GP(n, k))$ which, for moderate values of k, is tighter than the general upper bounds provided by Salazar in [123]. This appears to be a fertile area for future research.

In Section 4.5, we exhibited a function $f(n)$ which closely follows our best found heuristic solutions to the family of Sheehan graphs. Finding a drawing procedure for these graphs that gives $f(n)$ crossings (or something similar) is a topic for future research. In general, it would be interesting to study the least dense graphs on n vertices which appear to have crossing number of $\Omega(n^4)$. A related, and tantalisingly open, question asks whether there exists a family of cubic graphs on n vertices whose crossing number is $\Omega(n^2)$.

Appendix A

Counting crossings in a drawing

We now count the crossings of the drawing construction in Theorem 3.23. For convenience, we replicate the figure from that theorem here as Figure A.1. The number of crossings in all of the constructions throughout Section 3.3 can be determined with similar arguments.

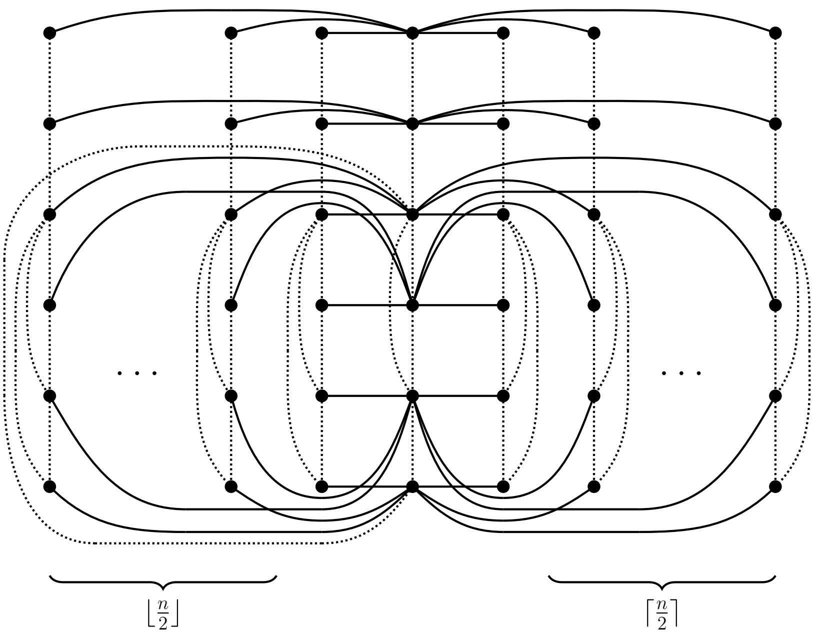


Figure A.1: $G^6_{62}\square S_n$ drawn with $\left\lfloor \frac{n}{2} \right\rfloor \left(5 \left\lfloor \frac{n-1}{2} \right\rfloor + 2\right)$ crossings. The solid lines are the edges of copies of S_n and the dashed lines are the edges of copies of G^6_{62}.

Figure A.1 indicates how $G^6_{62} \Box S_n$ should be drawn for arbitrarily large n. Note that if n is odd, then more copies of G^6_{62} are drawn on the right. Now, we seek to determine the number of crossings contained in Figure A.1 for general n.

Lemma A.1. *The number of crossings in the drawing construction of Figure A.1 is* $5 \left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2 \left\lfloor \frac{n}{2} \right\rfloor$.

Proof. Note that the solid lines are the edges of copies of S_n and the dashed lines are the edges of copies of G^6_{62}. In effect, each of the dashed vertical lines (along with attached dashed edges) corresponds to one copy of G^6_{62}. Also, the vertices lying on the vertical line in the centre of Figure A.1 correspond to the centre vertices in the copies of S_n. Now, we begin by counting the crossings on edges belonging to the copies of S_n. Consider the copies of G^6_{62} moving away to the left from the centre copy, and consider specifically those dashed edges which are drawn as straight vertical lines. It can be seen that as we move to the left, each copy is crossed by three fewer solid edges than the previous copy. This continues until the final copy of G^6_{62} for which there are no crossings involving the solid edges. The identical situation also occurs as we move to the right. Hence, the number of crossings between the solid edges and any vertical dashed edge is

$$3 \sum_{i=1}^{\left\lfloor \frac{n}{2} \right\rfloor - 1} i + 3 \sum_{i=1}^{\left\lceil \frac{n}{2} \right\rceil - 1} i.$$

Some of the edges of the copies of S_n also cross each other, and the number of these can also be counted:

$$2 \sum_{i=1}^{\left\lfloor \frac{n}{2} \right\rfloor - 1} i + 2 \sum_{i=1}^{\left\lceil \frac{n}{2} \right\rceil - 1} i.$$

Finally, the long looping dashed edge on the left of the diagram which begins and ends at a central vertex, crosses $\left\lfloor \frac{n}{2} \right\rfloor$ edges, and the shorter dashed edge on the left of the diagram which begins and ends at a central vertex, crosses $\left\lfloor \frac{n}{2} \right\rfloor$ edges. Putting all of the above together gives the total number of crossings

$$5\Big(\sum_{i=1}^{\lfloor \frac{n}{2} \rfloor - 1} i + \sum_{i=1}^{\lceil \frac{n}{2} \rceil - 1} i \Big) + 2 \left\lfloor \frac{n}{2} \right\rfloor.$$

It can be checked that the sums inside the brackets are equal to $\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor$ and thus the drawing construction provides

$$5\Big(\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor \Big) + 2 \left\lfloor \frac{n}{2} \right\rfloor$$

crossings, which is the desired number. $\square$

Appendix B

Proof of Theorem 4.19

We now provide the full proof of Theorem 4.19. Recall that beginning with the graph K_{n+d}, the edges of a vertex induced subgraph formed by any set of d vertices can be deleted and the resulting graph is $K_n + D_d$. Beginning with the particular kind of cylinder drawing of K_{n+d} described in Section 4.2, we labelled the vertices on the inner circle in a anticlockwise fashion as $a_1, a_2, \ldots, a_{\lceil (n+d)/2 \rceil}$. Then, beginning at the vertex immediately anticlockwise from the top vertex on the outside circle (if $n+d$ is even), or the vertex immediately anticlockwise from the top spot which is missing a vertex (if $n+d$ is odd), the vertices on the outside circle were labelled in an anticlockwise fashion as $b_1, b_2, \ldots, b_{\lfloor (n+d)/2 \rfloor}$. The edges of the vertex induced subgraph formed by the vertices $\{a_1, a_2, \ldots, a_{\lceil d/2 \rceil}\} \cup \{b_1, b_2, \ldots, b_{\lfloor d/2 \rfloor}\}$ are deleted to reduce the drawing to a *spiral cylinder drawing* of $K_n + D_d$.

Theorem B.1. *For $n + d \geq 6$, the number of crossings in a spiral cylinder drawing of $K_n + D_d$ is $Z(n, d) + H(n) + f(n, d)$ where,*

$$f(n, d) = \frac{1}{2} \left\lfloor \frac{nd}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor \left\lfloor \frac{n-2}{2} \right\rfloor .$$

Proof. Denote the vertex set $A := \{a_1, a_2, \ldots, a_{\lceil d/2 \rceil}\} \cup \{b_1, b_2, \ldots, b_{\lfloor d/2 \rfloor}\}$. Because the drawing changes slightly depending on the parities of n and d, boundary values in our counting must be handled carefully. Figure B.2

displays drawings and vertex indices for the different parities of n and d and Figure B.3 displays example drawings of $K_3 + D_4$ and $K_4 + D_4$. The (remaining) edges in the inner and outer regions are omitted for clarity. For the different parities of n and d, $f(n, d)$ simplifies to the following formulas.

parity	***f(n,d)***
n even, d even	$\frac{1}{16}dn^2(n-2)$
n odd, d odd	$\frac{1}{16}(dn-1)(n-1)(n-3)$
n even, d odd	$\frac{1}{16}dn^2(n-2)$
n odd, d even	$\frac{1}{16}dn(n-1)(n-3)$

Table B.1: $f(n, d)$ for different parities of n and d.

The crossings which are removed from the original drawing of K_{n+d} can be partitioned into six types. The first five types are a crossing between two edges (a_i, b_j) and (a_k, b_ℓ). The last type is a crossing in the inner circle or unbounded region, that is, a crossing between two edges (a_i, a_j) and (a_k, a_ℓ) or a crossing between two edges (b_i, b_j) and (b_k, b_ℓ). To avoid double counting, in the first four types of crossings, we will assume that $i > k$. So for any given crossing involving at least one edge whose end-vertices are both in A, exactly one of the following is true:

1. $a_i, b_j, a_k, b_\ell \in A$, where $i > j$ and $k > \ell$.
2. $a_i, b_j \notin A$, $a_k, b_\ell \in A$, where $i \leq j$ and $k > \ell$.
3. $b_j, a_k, b_\ell \in A$, $a_i \notin A$ where $k > \ell$.
4. $a_i, a_k, b_\ell \in A$, $b_j \notin A$ where $k > \ell$.
5. $a_i, b_j \in A$ where $i \leq j$.
6. The crossing occurs in either the inner circle or the unbounded region.

These six types of crossings are illustrated in Figure B.4 and they exhaust all possibilities for crossings which are removed from the original drawing. To see this, consider any edge $e = (a_r, b_s)$ where $a_r, b_s \in A$. Then e can possibly cross another edge whose end-vertices are both in A, and in this case, such a

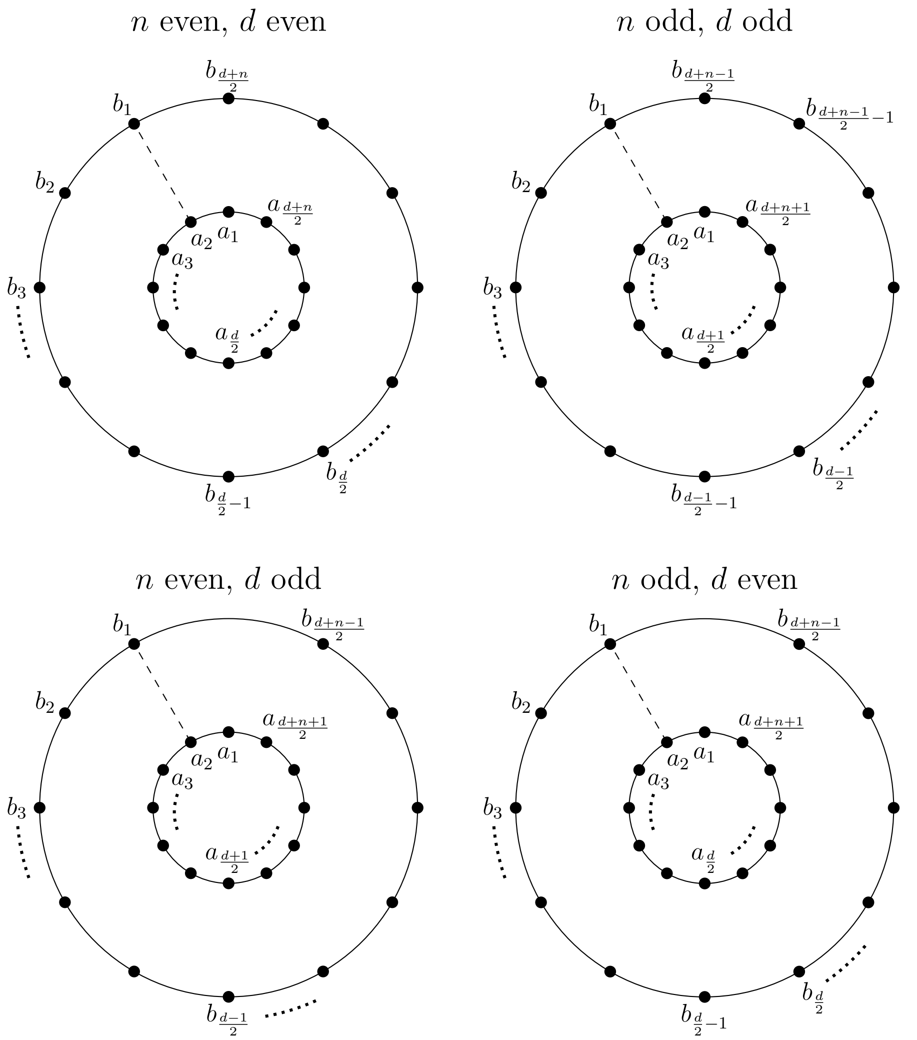


Figure B.2: The four possible vertex and index layouts, depending on the parities of n and d.

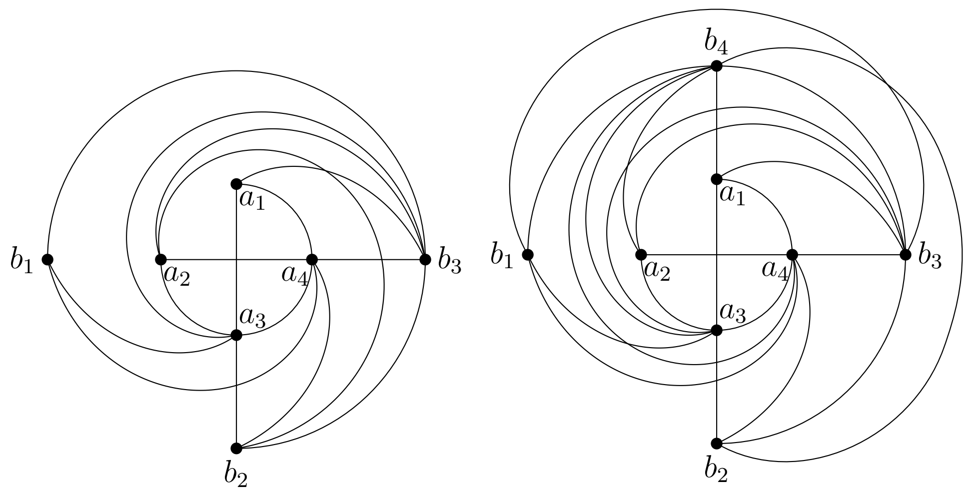


Figure B.3: On the left, a spiral cylinder drawing of $K_3 + D_4$ and on the right, a spiral cylinder drawing of $K_4 + D_4$.

crossing is one of either type 1 or type 5. Edge e can also possibly cross an edge which has one end-vertex in A and the other not in A, and in this case, such a crossing is exactly one of either type 3, type 4 or type 5. Finally, e can possibly cross an edge whose end-vertices both are not in A, and in this case, such a crossing is one of either type 2 or type 5. Next, consider an edge $e = (a_r, a_s)$ where $a_r, a_s \in A$. Then any crossing on e is of type 6. Similarly for the crossings of an edge $e = (b_r, b_s)$ where $b_r, b_s \in A$. We now consider each type of crossing individually, and compute the number of them.

Crossings of type 1. In this case we have $a_i, b_j, a_k, b_\ell \in A$ and $i > j$, $k > \ell$. We will consider each edge (a_i, b_j) separately and determine how many crossings exist with edges (a_k, b_ℓ), satisfying $a_k, b_\ell \in A$ and $k > \ell$. It was mentioned above that in order to avoid double counting, we assume that $i > k$. Also, we have $\ell > j$,or else the two edges do not cross. Hence, we have $i > k > \ell > j$. Note that this implies $i \geq j + 3$. Then for each i and j, we have a crossing for each k, ℓ satisfying $i > k > \ell > j$. There are $\binom{i-j-1}{2}$ such choices of k and ℓ. Hence, the number of crossings of the first type is:

$$f_1 := \sum_{i=4}^{\lceil \frac{d}{2} \rceil} \sum_{j=1}^{i-3} \binom{i-j-1}{2}.$$

Crossings of type 2. In this case we have $a_i, b_j \notin A$, $a_k, b_\ell \in A$ and $i \leq j$ and $k > \ell$. We will consider each edge (a_i, b_j) separately and determine which crossings exist with edges (a_k, b_ℓ), satisfying $a_k, b_\ell \in A$ and $k > \ell$. Given an edge (a_i, b_j) which satisfies $a_i, b_j \notin A$ and $i \leq j$, it forms a crossing with every edge (a_k, b_ℓ) where $a_k, b_\ell \in A$ and $k > \ell$. Because k and ℓ must satisfy $1 \leq \ell < k \leq \lceil d/2 \rceil$, there are $\binom{\lceil d/2 \rceil}{2}$ such edges (a_k, b_ℓ). For the edges (a_i, b_j), i and j must satisfy $\lceil d/2 \rceil + 1 \leq i \leq j \leq \lfloor (n+d)/2 \rfloor$, hence there are

$$\binom{\lfloor (n+d)/2 \rfloor - \lceil d/2 \rceil + 1}{2}$$

such edges. Hence the following calculation gives the number of crossings of

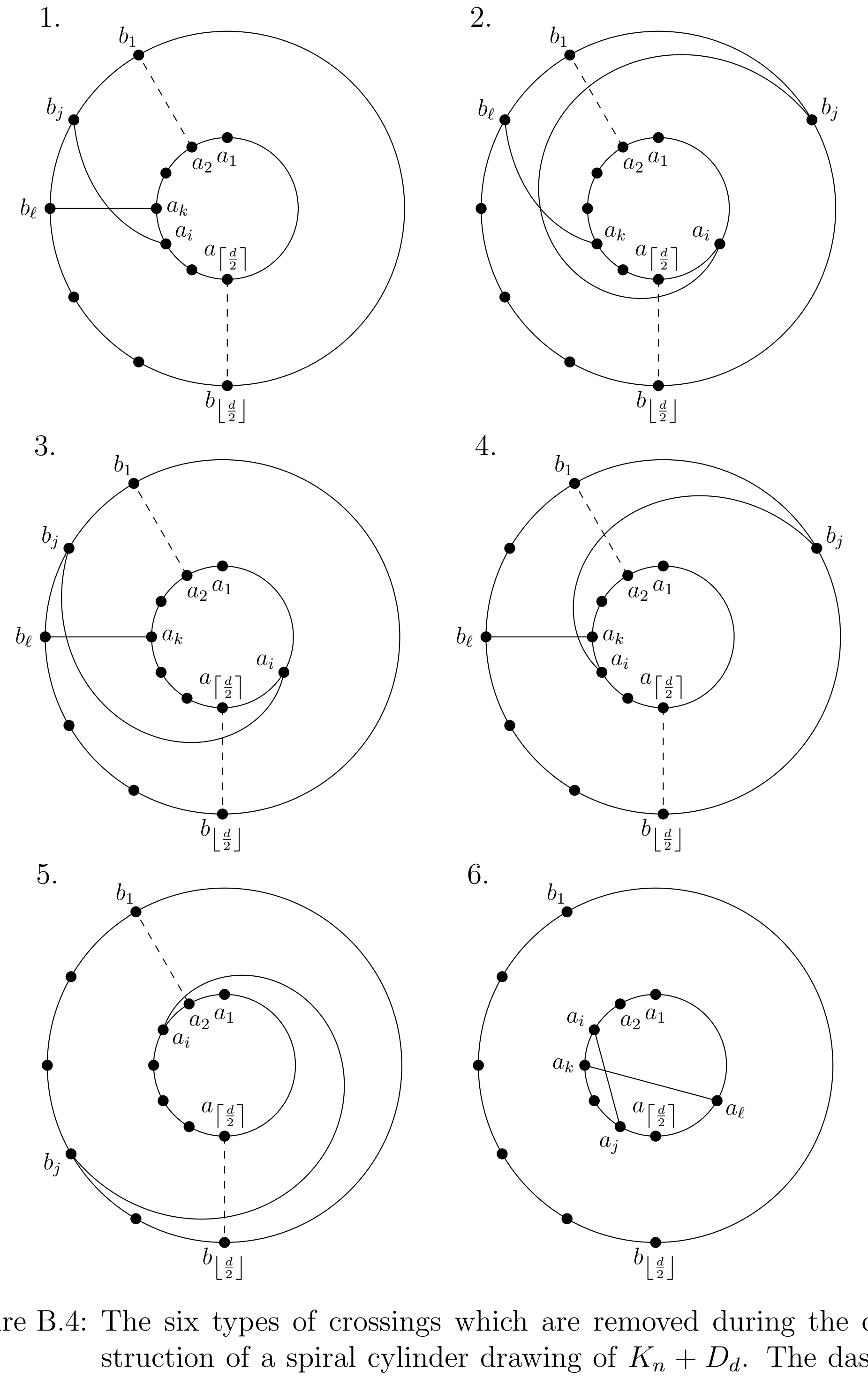


Figure B.4: The six types of crossings which are removed during the construction of a spiral cylinder drawing of $K_n + D_d$. The dashed lines give an example of the correspondence, in the drawing, between the a vertices and the b vertices.

the second type:

$$f_2 := \binom{\lfloor (n+d)/2 \rfloor - \lceil d/2 \rceil + 1}{2} \binom{\lceil d/2 \rceil}{2}.$$

Crossings of type 3. In this case we have $b_j, a_k, b_\ell \in A$, $a_i \notin A$ and $k > \ell$. We will consider each edge (a_i, b_j) separately and determine which crossings exist with edges (a_k, b_ℓ), satisfying $a_k, b_\ell \in A$ and $k > \ell$. Consider an edge (a_i, b_j) where $a_i \notin A$, $b_j \in A$. Then this edge forms a crossing with another edge (a_k, b_ℓ) where $a_k, b_\ell \in A$ and $k > \ell$, if and only if $j < \ell$ (and hence $k > j+1$). Because k and ℓ must also satisfy $\lceil d/2 \rceil \geq k > \ell \geq 1$, there are $\binom{\lceil d/2 \rceil - j}{2}$ ways to choose such an edge (a_k, b_ℓ). Note that this also implies that $j \leq \lceil d/2 \rceil - 2$. Then, for each j, in order to satisfy $a_i \notin A$, i satisfies $\lceil d/2 \rceil + 1 \geq i \geq \lceil (d+n)/2 \rceil$, of which there are $\lceil (d+n)/2 \rceil - \lceil d/2 \rceil$ possibilities. Hence, the following calculation gives the number of crossings of the third type:

$$f_3 := \sum_{j=1}^{\lceil d/2 \rceil - 2} (\lceil (d+n)/2 \rceil - \lceil d/2 \rceil) \binom{\lceil d/2 \rceil - j}{2}.$$

Crossings of type 4. In this case we have $a_i, a_k, b_\ell \in A$, $b_j \notin A$ and $k > \ell$. Again, we will consider each edge (a_i, b_j) separately and determine which crossings exist with edges (a_k, b_ℓ), satisfying $a_k, b_\ell \in A$ and $k > \ell$. Consider an edge (a_i, b_j) where $a_i \in A$, $b_j \notin A$. Then this edge forms a crossing with another edge (a_k, b_ℓ) where $a_k, b_\ell \in A$ and $k > \ell$, whenever $i > k$ and $i - 1 > \ell$. Because k and ℓ must also satisfy $(d+1)/2 \geq k > \ell \geq 1$, there are $\binom{i-1}{2}$ ways to choose such an edge (a_k, b_ℓ). Note that this also implies that $i \geq 3$. Then, for each i, in order to satisfy $b_j \notin A$, j must satisfy $\lfloor d/2 \rfloor + 1 \leq j \leq \lfloor (n+d)/2 \rfloor$, of which there are $\lfloor (n+d)/2 \rfloor \lfloor d/2 \rfloor$ possibilities. Hence, the following calculation gives the number of crossings of the fourth type:

$$f_4 := \sum_{i=3}^{\lceil d/2 \rceil} \left(\lfloor (n+d)/2 \rfloor \lfloor d/2 \rfloor \right) \binom{i-1}{2}.$$

Crossings of type 5. In this case we have $a_i, b_j \in A$ and $i \leq j$. We will consider each edge (a_i, b_j) separately and determine which crossings exist with any edges (a_k, b_ℓ). Consider an edge (a_i, b_j) where $a_i, b_j \in A$ and $i \leq j$. Then this edge forms a crossing with an edge (a_k, b_ℓ) for particular values of k and ℓ. Specifically, k can be any one of

$$\{i-1, i-2, \ldots, 1, \lceil (d+n)/2 \rceil, \lceil (d+n)/2 \rceil - 1, \ldots, j+2\}.$$

Then, given such a k, if it lies within $\{1, 2, \ldots, i-1\}$ then a crossing exists whenever ℓ is any one of

$$\{k-1, k-2, \ldots, 1, \lfloor (d+n)/2 \rfloor, \lfloor (d+n)/2 \rfloor - 1, \ldots, j+1\}.$$

Otherwise, k lies within $\{j+2, \ldots, \lceil (d+n)/2 \rceil\}$ and a crossing exists whenever ℓ is any one of

$$\{k-1, k-2, \ldots, j+1\}.$$

There are $r := (i-1) + \lceil (d+n)/2 \rceil - (j+1)$ possible values for k, and because the a_k is connected differently here depending on whether $n+d$ is odd or even, we consider these cases separately.

Case where $n+d$ is even. In this case the edge (a_i, b_j) forms crossings with $\binom{r+1}{2}$ edges (a_k, b_ℓ). Therefore, the following calculation gives the number of crossings of the fifth type for $n+d$ even:

$$\sum_{i=1}^{\lceil d/2 \rceil} \sum_{j=i}^{\lfloor d/2 \rfloor} \binom{r+1}{2}.$$

Case where $n+d$ odd. In this case, a_1 has no corresponding vertex on the outside circle, and so the edge (a_i, b_j) forms fewer than $\binom{r+1}{2}$ crossings with edges (a_k, b_ℓ). Specifically, the edge (a_i, b_j) forms crossings with

$$\binom{r}{2} + \lfloor (d+n)/2 \rfloor - j$$

edges (a_k, b_ℓ). Therefore, the following calculation gives the number of crossings of the fifth type for $n + d$ odd:

$$\sum_{i=1}^{\lceil d/2 \rceil} \sum_{j=i}^{\lfloor d/2 \rfloor} \binom{r}{2} + \lfloor (d+n)/2 \rfloor - j.$$

Let f_5 be the function resulting from whichever of the two cases is appropriate.

Crossings of type 6. In this case, we consider any crossings which occur in the inner circle, or in the unbounded region. We first consider the inner circle and count the crossings on each of the edges which are removed. There are $\binom{\lceil d/2 \rceil}{4}$ crossings being removed which involve edges (a_i, a_j) and (a_k, a_ℓ) where $a_i, a_j, a_k, a_\ell \in A$. Next, given an edge (a_i, a_j) with $a_i, a_j \in A$, for each $k \in \{i+1, i+2, \ldots, j-1\}$, there exists $\lceil (d+n)/2 \rceil - \lceil d/2 \rceil$ crossings with edges of the form (a_k, a_ℓ) where $a_\ell \notin A$. Therefore the following calculation gives the number of crossings being removed from the inner circle:

$$f_{6a} := \binom{\lceil d/2 \rceil}{4} + \sum_{i=1}^{\lceil d/2 \rceil} \sum_{j=i+2}^{\lceil d/2 \rceil} (j-i-1)(\lceil (d+n)/2 \rceil - \lceil d/2 \rceil).$$

Using an analogous argument, the number of crossings removed from the unbounded region is:

$$f_{6b} := \binom{\lfloor d/2 \rfloor}{4} + \sum_{i=1}^{\lfloor d/2 \rfloor} \sum_{j=i+2}^{\lfloor d/2 \rfloor} (j-i-1)(\lfloor (d+n)/2 \rfloor - \lfloor d/2 \rfloor).$$

Let $f_6 = f_{6a} + f_{6b}$.

We have shown that the number of crossings in a spiral cylinder drawing of $K_n + D_d$ is

$$H(n+d) - \sum_{i=1}^{6} f_i.$$

Converting into an expression involving $H(n)$ and $Z(n,d)$, we have

$$Z(n,d) + H(n) + F = H(n+d) - \sum_{i=1}^{6} f_i,$$

and therefore,

$$F = H(n+d) - \sum_{i=1}^{6} f_i - Z(n,d) - H(n). \tag{B.1}$$

The final task is to prove that (B.1) coincides with the expression in Theorem B.1. In order to evaluate (B.1) it is worthwhile removing the ceilings and floors so that the various f_i can be combined into single expressions. This requires consideration of the parity of both n and d, and hence there are four cases to consider. To complete the proof, in each of the four cases, we used a computer aided simplification tool to confirm that F coincides with the simplifications given in Table B.1. $\square$

Appendix C

Predicted crossing numbers of graphs resulting from products

In Section 3.3, we studied the Cartesian products of fixed small graphs with arbitrarily large paths, cycles or stars and the join products of fixed small graphs with arbitrarily large discrete graphs, paths or cycles. For many of the possible small fixed graphs, the crossing numbers of the resulting family have not yet been determined and for these cases, we now attempt to predict what the crossing number should be. To achieve this, we considered all small graphs on five and six vertices, and ran the first ten instances of the resulting families using Quickcross. We then observed the results and predicted the formula for the crossing numbers of each family. At the end of this process, we observed that for all of the cases with known crossing numbers, our predicted formulas coincided with the crossing numbers. This then leads us to conjecture that, for the unknown cases, our predicted formulas will also coincide with the crossing number.

To obtain the predictions, each considered instance was run with Quickcross many thousands of times to ensure that we are unable to find a solution with fewer crossings. Each of our predicted formulas holds for values of n which are 'large enough' and although this is often simply $n \geq 1$, or $n \geq 3$

for cycles, sometimes these bounds are different. We omit these numbers because they are hard to display in their entirety, however if desired, they can be easily determined by running the instances again with Quickcross. Although some of our predicted formulas can be written nicer, we display them in the most pleasing general formula that we found, which we detail now.

For Cartesian products involving paths of length n, we observed that in all known cases, the formula is linear in n. For Cartesian products involving cycles of length n, we rely on the prediction from Section 4.1 which indicates that the formula should be linear in n, but may change depending on the parity of n. Hence, some of these formulas involve floors. For Cartesian products involving stars of size n, we observe that the general formula is as follows

$$a\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+bn+c\left\lfloor\frac{n}{2}\right\rfloor+d, \tag{C.1}$$

for some integers a, b, c, d. For all of the join products, the formula also obeys (C.1), with the additional requirement if the fixed graph is of order m, then $a=\left\lfloor\frac{m}{2}\right\rfloor\left\lfloor\frac{m-1}{2}\right\rfloor$. Then, in each case, in order to predict the formula, we simply look at the number of unknowns in the predicted general formula, and compare it to that many results from Quickcross to determine the coefficients. Finally, we check this result by seeing if it predicted the remaining found values from Quickcross.

Tables C.1 and C.2 display our predictions for when the fixed small graph is a five vertex graph, then Tables C.3 and C.4 display our predictions when the fixed small graph is a six vertex graph. We include both the known results, and our predictions; the latter cases are highlighted in green. Note that the newly decided cases from Section 3.3 are included in the following tables as known results.

Table C.1: Predicted crossing numbers for the Cartesian product of a five vertex graph with cycles and stars.

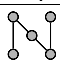
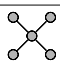
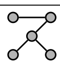
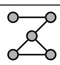
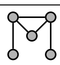
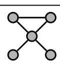
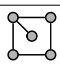
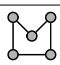
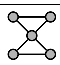
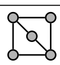
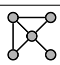
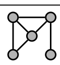
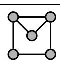
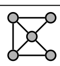
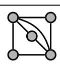
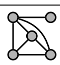
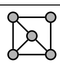
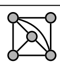
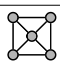
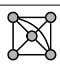
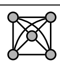
i	G_i^5	$G_i^5 \Box P_n$	$G_i^5 \Box C_n$	$G_i^5 \Box S_n$
1		0	0	$3\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor$
2		$2n-2$	$2n$	$n(n-1)$
3		$n-1$	n	$3\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+\left\lfloor\frac{n}{2}\right\rfloor$
4		$n-1$	n	$3\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+\left\lfloor\frac{n}{2}\right\rfloor$
5		$n-1$	n	$Z(5,n)+\left\lfloor\frac{n}{2}\right\rfloor$
6		$2n-2$	$2n$	$n(n-1)$
7		$n-1$	$2n$	$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+\left\lfloor\frac{n}{2}\right\rfloor$
8		0	$3n$	$Z(5,n)$
9		$2n-2$	$2n$	$n(n-1)$
10		$2n$	$4n$	$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n$
11		$2n-2$	$3n$	$n(n-1)$
12		$2n-2$	$2n$	$n(n-1)$
13		$n-1$	$3n$	$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+\left\lfloor\frac{n}{2}\right\rfloor$
14		$2n-2$	$3n$	$n(n-1)$
15		$3n-1$	$5n$	$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+\left\lfloor\frac{n}{2}\right\rfloor$
16		$3n-1$	$4n-2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(5,n)+2n+\left\lfloor\frac{n}{2}\right\rfloor$
17		$2n$	$5n-2\left\lfloor\frac{n}{2}\right\rfloor$	$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n$
18		$3n-1$	$5n$	$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+\left\lfloor\frac{n}{2}\right\rfloor$
19		$3n-1$	$6n-4\left\lfloor\frac{n}{2}\right\rfloor$	$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+\left\lfloor\frac{n}{2}\right\rfloor$
20		$4n$	$6n$	$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n$
21		$6n$	$9n$	$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+5n+\left\lfloor\frac{n}{2}\right\rfloor+1$

Table C.2: Predicted crossing numbers for the join product of a five vertex graph with discrete graphs, paths and cycles

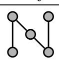
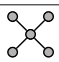
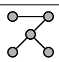
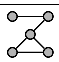
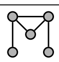
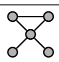
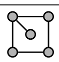
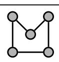
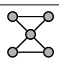
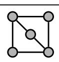
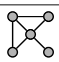
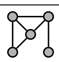
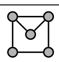
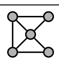
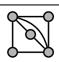
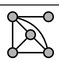
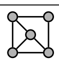
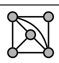
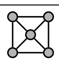
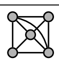
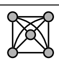
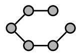
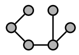
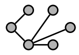
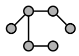
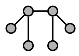
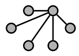
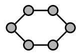
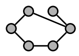
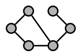
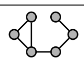
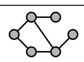
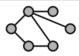
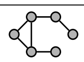
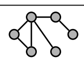
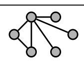
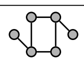
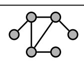
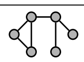
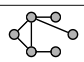
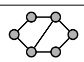
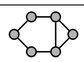
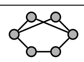
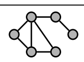
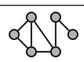
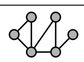
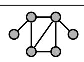
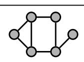
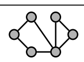
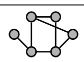
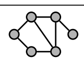
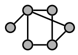
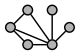
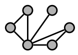
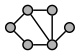
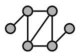
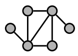
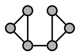
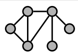
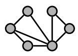
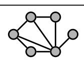
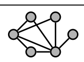
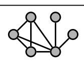
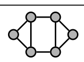
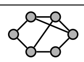
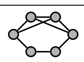
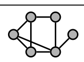
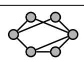
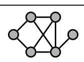
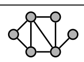
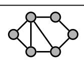
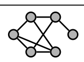
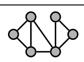
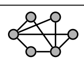
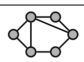
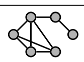
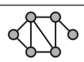
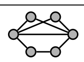
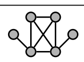
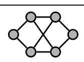
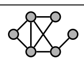
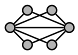
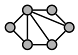
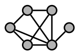
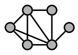
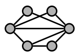
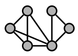
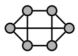
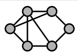
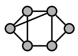
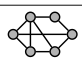
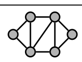
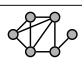
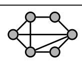
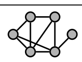
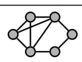
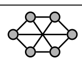
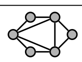
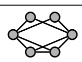
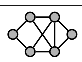
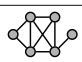
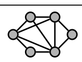
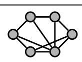
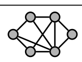
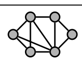
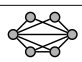
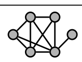
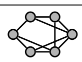
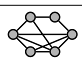
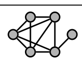
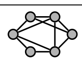
i	G_i^5	$G_i^5 + D_n$	$G_i^5 + P_{n-1}$	$G_i^5 + C_n$
1		$Z(5,n)$	$Z(5,n)$	$Z(5,n)+1$
2		$n(n-1)$	$n(n-1)$	$n(n-1)+2$
3		$Z(5,n)+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+\lfloor\frac{n}{2}\rfloor+1$
4		$Z(5,n)+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+\lfloor\frac{n}{2}\rfloor+1$
5		$Z(5,n)+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+\lfloor\frac{n}{2}\rfloor+2$
6		$Z(5,n)+2\lfloor\frac{n}{2}\rfloor$	$n(n-1)$	$n(n-1)+2$
7		$Z(5,n)+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+\lfloor\frac{n}{2}\rfloor+1$	$Z(5,n)+\lfloor\frac{n}{2}\rfloor+2$
8		$Z(5,n)$	$Z(5,n)+1$	$Z(5,n)+2$
9		$Z(5,n)+2\lfloor\frac{n}{2}\rfloor$	$n(n-1)$	$n(n-1)+2$
10		$Z(5,n)+n$	$Z(5,n)+n+1$	$Z(5,n)+n+3$
11		$n(n-1)$	$n(n-1)+1$	$n(n-1)+3$
12		$n(n-1)$	$n(n-1)$	$Z(5,n)+2\lfloor\frac{n}{2}\rfloor+3$
13		$Z(5,n)+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+\lfloor\frac{n}{2}\rfloor+1$	$Z(5,n)+\lfloor\frac{n}{2}\rfloor+2$
14		$n(n-1)$	$n(n-1)+1$	$n(n-1)+3$
15		$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor+2$	$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor+4$
16		$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor+1$	$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor+3$
17		$Z(5,n)+n$	$Z(5,n)+n+1$	$Z(5,n)+n+3$
18		$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor+2$	$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor+4$
19		$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor$	$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor+1$	$Z(5,n)+n+\lfloor\frac{n}{2}\rfloor+4$
20		$Z(5,n)+2n$	$Z(5,n)+2n+2$	$Z(5,n)+2n+5$
21		$Z(5,n)+2n+\lfloor\frac{n}{2}\rfloor+1$	$Z(5,n)+2n+\lfloor\frac{n}{2}\rfloor+4$	$Z(5,n)+3n-\lfloor\frac{n}{2}\rfloor+4$

Table C.3: Predicted crossing numbers for the Cartesian product of a six vertex graph with paths, cycles and stars.

i	G_i^6	$G_i^6 \square P_n$	$G_i^6 \square C_n$	$G_i^6 \square S_n$
25		0	0	$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor$
26		$n-1$	n	$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+\left\lfloor\frac{n}{2}\right\rfloor$
27		$2n-2$	$2n$	$5\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$
28		$n-1$	n	$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+\left\lfloor\frac{n}{2}\right\rfloor$
29		$2n-2$	$2n$	$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$
31		$4n-4$	$4n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4\left\lfloor\frac{n}{2}\right\rfloor$
40		0	$4n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor$
41		$n-1$	$3n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+\left\lfloor\frac{n}{2}\right\rfloor$
42		$2n-4$	$2n$	$5\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+\left\lfloor\frac{n}{2}\right\rfloor$
43		$n-1$	n	$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+\left\lfloor\frac{n}{2}\right\rfloor$
44		$2n-2$	$2n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$
45		$2n-2$	$3n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$
46		$n-1$	n	$5\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+\left\lfloor\frac{n}{2}\right\rfloor$
47		$2n-2$	$2n$	$5\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$
48		$4n-4$	$4n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4\left\lfloor\frac{n}{2}\right\rfloor$
49		$2n-2$	$2n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$
51		$3n-3$	$3n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+3\left\lfloor\frac{n}{2}\right\rfloor$
53		$2n-2$	$2n$	$4\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$
54		$2n-2$	$2n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$
59		$2n-2$	$4n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$
60		$n-1$	$4n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+\left\lfloor\frac{n}{2}\right\rfloor$
61		$2n$	$5n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n$
62		$3n-5$	$3n$	$5\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$
63		$2n-2$	$2n$	$5\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$
64		$2n-2$	$2n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$
65		$3n-3$	$3n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+3\left\lfloor\frac{n}{2}\right\rfloor$
66		$2n-2$	$3n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2\left\lfloor\frac{n}{2}\right\rfloor$
67		$3n-3$	$3n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+3\left\lfloor\frac{n}{2}\right\rfloor$
68		$3n-1$	$4n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+\left\lfloor\frac{n}{2}\right\rfloor$
70		$3n-3$	$3n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+3\left\lfloor\frac{n}{2}\right\rfloor$

i	G_i^6	$G_i^6 \square P_n$	$G_i^6 \square C_n$	$G_i^6 \square S_n$
71		$3n-1$	$5n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2n + \left\lfloor \frac{n}{2} \right\rfloor$
72		$4n-4$	$4n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 4\left\lfloor \frac{n}{2} \right\rfloor$
73		$4n-4$	$4n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 4\left\lfloor \frac{n}{2} \right\rfloor$
74		$2n-2$	$4n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2\left\lfloor \frac{n}{2} \right\rfloor$
75		$2n$	$2n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 4\left\lfloor \frac{n}{2} \right\rfloor$
76		$3n-2$	$4n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 3\left\lfloor \frac{n}{2} \right\rfloor$
77		$2n-2$	$2n$	$4\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2\left\lfloor \frac{n}{2} \right\rfloor$
78		$3n-3$	$3n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 3\left\lfloor \frac{n}{2} \right\rfloor$
79		$4n-4$	$4n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 4\left\lfloor \frac{n}{2} \right\rfloor$
80		$4n-4$	$5n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 4\left\lfloor \frac{n}{2} \right\rfloor$
81		$5n-2$	$6n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2n + 3\left\lfloor \frac{n}{2} \right\rfloor$
82		$4n-2$	$3n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 3n + \left\lfloor \frac{n}{2} \right\rfloor$
83		$2n-2$	$4n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2\left\lfloor \frac{n}{2} \right\rfloor$
84		$3n-1$	$6n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2n + \left\lfloor \frac{n}{2} \right\rfloor$
85		$2n$	$4n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2n$
86		$3n-1$	$5n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2n + \left\lfloor \frac{n}{2} \right\rfloor$
87		$3n-1$	$5n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2n + \left\lfloor \frac{n}{2} \right\rfloor$
88		$4n-2$	$5n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor$
89		$3n-3$	$4n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 3\left\lfloor \frac{n}{2} \right\rfloor$
90		$3n-3$	$4n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 3\left\lfloor \frac{n}{2} \right\rfloor$
91		$3n-1$	$3n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2n + \left\lfloor \frac{n}{2} \right\rfloor$
92		$3n-3$	$3n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 4\left\lfloor \frac{n}{2} \right\rfloor$
93		$4n$	$8n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 4n$
94		$2n-2$	$5n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2\left\lfloor \frac{n}{2} \right\rfloor$
95		$3n-1$	$2n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$5\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2n + \left\lfloor \frac{n}{2} \right\rfloor$
98		$3n-1$	$3n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 4\left\lfloor \frac{n}{2} \right\rfloor$
99		$4n-2$	$6n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor$
100		$4n-2$	$4n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor$
101		$4n-2$	$7n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor$
102		$5n-3$	$5n$	$6\left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor + 1n + 5\left\lfloor \frac{n}{2} \right\rfloor$

i	G_i^6	$G_i^6 \square P_n$	$G_i^6 \square C_n$	$G_i^6 \square S_n$
103		$6n-2$	$8n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n+2\left\lfloor\frac{n}{2}\right\rfloor$
104		$4n-4$	$5n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4\left\lfloor\frac{n}{2}\right\rfloor$
105		$6n-2$	$7n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n+2\left\lfloor\frac{n}{2}\right\rfloor$
106		$5n-3$	$6n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+3\left\lfloor\frac{n}{2}\right\rfloor$
107		$5n-3$	$6n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+3\left\lfloor\frac{n}{2}\right\rfloor$
108		$4n-2$	$3n+2\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+3n+\left\lfloor\frac{n}{2}\right\rfloor$
109		$4n$	$4n+4\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n$
110		$3n-1$	$5n+2\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+\left\lfloor\frac{n}{2}\right\rfloor$
111		$3n-1$	$4n+2\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+\left\lfloor\frac{n}{2}\right\rfloor$
112		$4n$	$7n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n$
113		$4n-4$	$4n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4\left\lfloor\frac{n}{2}\right\rfloor$
114		$5n-3$	$5n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+3\left\lfloor\frac{n}{2}\right\rfloor$
115		$4n-2$	$6n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+2\left\lfloor\frac{n}{2}\right\rfloor$
116		$4n-2$	$4n+2\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+2\left\lfloor\frac{n}{2}\right\rfloor$
118		$4n-2$	$6n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+2\left\lfloor\frac{n}{2}\right\rfloor$
119		$7n-1$	$9n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+5n+2\left\lfloor\frac{n}{2}\right\rfloor+1$
120		$3n-3$	$5n+2\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+3\left\lfloor\frac{n}{2}\right\rfloor$
121		$4n$	$8n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n$
122		$5n-3$	$7n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+3\left\lfloor\frac{n}{2}\right\rfloor$
123		$5n-3$	$5n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+3\left\lfloor\frac{n}{2}\right\rfloor$
124		$5n-3$	$7n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+3\left\lfloor\frac{n}{2}\right\rfloor$
125		$5n-3$	$3n+6\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+3\left\lfloor\frac{n}{2}\right\rfloor$
126		$6n-2$	$6n+2\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n+2\left\lfloor\frac{n}{2}\right\rfloor$
127		$5n-3$	$6n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+3\left\lfloor\frac{n}{2}\right\rfloor$
128		$6n-1$	$8n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n+2\left\lfloor\frac{n}{2}\right\rfloor$
129		$6n-2$	$6n+2\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n+2\left\lfloor\frac{n}{2}\right\rfloor$
130		$4n$	$6n+2\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n$
131		$5n-1$	$5n+4\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n+\left\lfloor\frac{n}{2}\right\rfloor$
132		$6n-2$	$4n+4\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n+2\left\lfloor\frac{n}{2}\right\rfloor$
133		$4n-2$	$5n+2\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+2\left\lfloor\frac{n}{2}\right\rfloor$

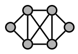
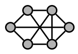
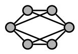
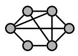
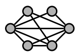
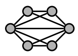
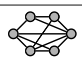
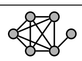
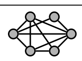
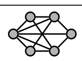
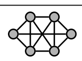
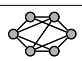
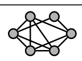
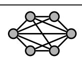
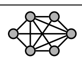
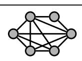
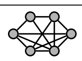
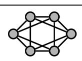
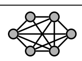
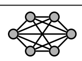
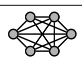
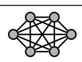
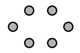
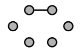
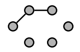
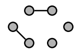
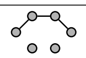
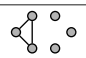
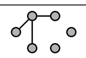
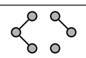
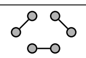
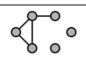
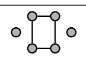
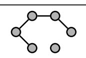
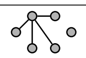
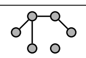
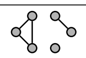
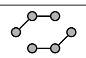
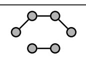
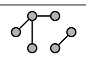
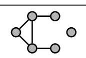
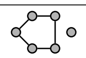
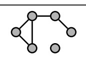
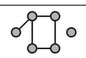
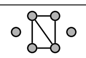
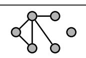
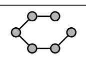
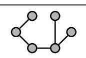
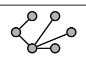
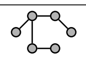
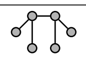
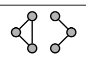
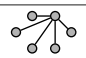
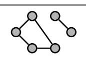
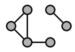
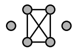
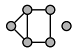
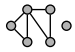
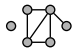
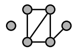
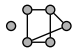
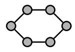
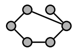
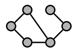
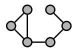
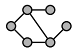
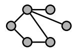
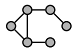
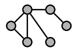
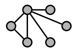
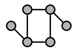
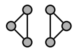
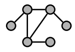
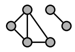
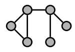
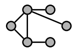
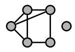
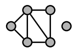
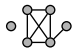
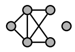
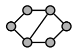
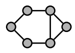
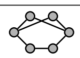
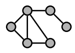
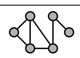
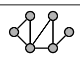
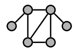
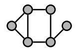
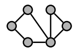
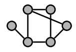
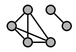
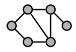
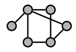
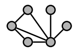
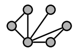
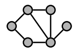
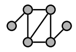
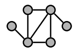
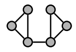
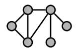
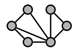
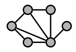
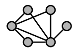
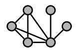
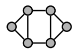
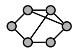
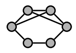
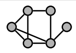
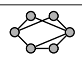
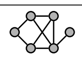
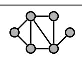
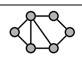
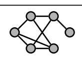
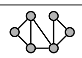
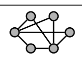
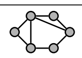
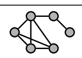
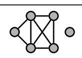
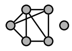
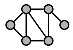
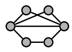
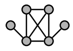
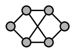
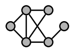
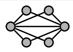
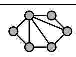
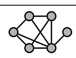
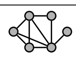
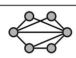
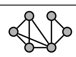
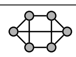
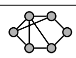
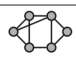
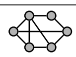
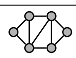
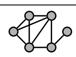
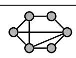
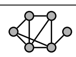
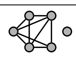
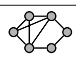
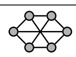
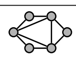
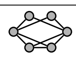
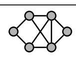
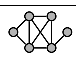
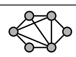
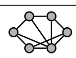
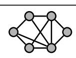
i	G_i^6	$G_i^6 \square P_n$	$G_i^6 \square C_n$	$G_i^6 \square S_n$
134		$6n-4$	$7n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+2n+4\left\lfloor\frac{n}{2}\right\rfloor$
135		$7n-1$	$8n+2\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+5n+2\left\lfloor\frac{n}{2}\right\rfloor+1$
137		$4n$	$8n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n$
138		$5n-1$	$7n+2\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n+\left\lfloor\frac{n}{2}\right\rfloor$
139		$7n-1$	$9n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+6n+\left\lfloor\frac{n}{2}\right\rfloor$
140		$6n-2$	$8n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n+2\left\lfloor\frac{n}{2}\right\rfloor$
141		$6n-1$	$4n+6\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n+2\left\lfloor\frac{n}{2}\right\rfloor$
142		$9n-3$	$9n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+5n+4\left\lfloor\frac{n}{2}\right\rfloor+1$
143		$6n-2$	$8n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n+2\left\lfloor\frac{n}{2}\right\rfloor$
144		$8n-1$	$10n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+5n+3\left\lfloor\frac{n}{2}\right\rfloor+1$
145		$7n-1$	$7n+4\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+5n+2\left\lfloor\frac{n}{2}\right\rfloor+1$
146		$5n-1$	$6n+4\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+4n+\left\lfloor\frac{n}{2}\right\rfloor$
147		$8n-2$	$10n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+5n+3\left\lfloor\frac{n}{2}\right\rfloor+1$
148		$7n-1$	$9n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+6n+\left\lfloor\frac{n}{2}\right\rfloor$
149		$10n$	$12n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+9n+\left\lfloor\frac{n}{2}\right\rfloor+1$
150		$9n-3$	$11n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+5n+4\left\lfloor\frac{n}{2}\right\rfloor+1$
151		$8n$	$8n+4\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+5n+3\left\lfloor\frac{n}{2}\right\rfloor+1$
152		$6n$	$12n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+6n$
153		$10n$	$12n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+9n+\left\lfloor\frac{n}{2}\right\rfloor+1$
154		$9n-1$	$11n+2\left\lfloor\frac{n}{2}\right\rfloor$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+7n+2\left\lfloor\frac{n}{2}\right\rfloor+1$
155		$12n$	$15n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+10n+2\left\lfloor\frac{n}{2}\right\rfloor+2$
156		$15n+3$	$18n$	$6\left\lfloor\frac{n}{2}\right\rfloor\left\lfloor\frac{n-1}{2}\right\rfloor+15n+3$

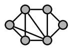
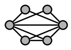
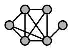
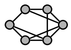
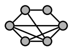
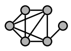
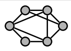
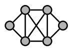
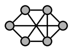
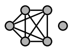
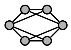
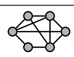
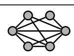
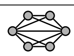
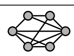
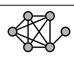
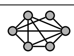
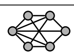
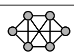
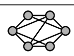
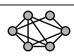
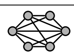
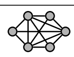
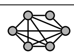
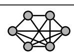
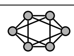
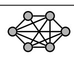
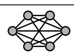
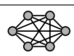
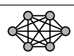
Table C.4: Predicted crossing numbers for the join product of a six vertex graph with discrete graphs, paths and cycles

i	G_i^6	$G_i^6 + D_n$	$G_i^6 + P_{n-1}$	$G_i^6 + C_n$
1		$Z(6,n)$	$Z(6,n)$	$Z(6,n)$
2		$Z(6,n)$	$Z(6,n)$	$Z(6,n)$
3		$Z(6,n)$	$Z(6,n)$	$Z(6,n)$
4		$Z(6,n)$	$Z(6,n)$	$Z(6,n)$
5		$Z(6,n)$	$Z(6,n)$	$Z(6,n)+1$
6		$Z(6,n)-n+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)-n+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)-n+3\left\lfloor\frac{n}{2}\right\rfloor$
7		$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor+1$
8		$Z(6,n)$	$Z(6,n)$	$Z(6,n)$
9		$Z(6,n)$	$Z(6,n)$	$Z(6,n)+1$
10		$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor+1$
11		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+2$
12		$Z(6,n)$	$Z(6,n)$	$Z(6,n)+1$
13		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+2$
14		$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor+1$
15		$Z(6,n)-n+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)-n+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)-n+3\left\lfloor\frac{n}{2}\right\rfloor$
16		$Z(6,n)$	$Z(6,n)$	$Z(6,n)$
17		$Z(6,n)$	$Z(6,n)$	$Z(6,n)+1$
18		$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor+1$
19		$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor+2$
20		$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor+2$
21		$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+2$
22		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+2$
23		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+2$
24		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+2$
25		$Z(6,n)$	$Z(6,n)$	$Z(6,n)+1$
26		$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor+1$
27		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+2$
28		$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+\left\lfloor\frac{n}{2}\right\rfloor+2$
29		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+1$
30		$Z(6,n)-n+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)-n+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)-n+3\left\lfloor\frac{n}{2}\right\rfloor$
31		$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor+3$
32		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+2$

i	G_i^6	$G_i^6 + D_n$	$G_i^6 + P_{n-1}$	$G_i^6 + C_n$
33		$Z(6,n) + \lfloor \frac{n}{2} \rfloor$	$Z(6,n) + \lfloor \frac{n}{2} \rfloor$	$Z(6,n) + \lfloor \frac{n}{2} \rfloor + 1$
34		$Z(6,n) + n + 2\lfloor \frac{n}{2} \rfloor - 1$	$Z(6,n) + 2n - 1$	$Z(6,n) + 2n + 2$
35		$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor + 1$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor + 2$
36		$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor + 2$
37		$Z(6,n) + 3\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 3\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 3\lfloor \frac{n}{2} \rfloor + 3$
38		$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor + 2$
39		$Z(6,n) + n + \lfloor \frac{n}{2} \rfloor$	$Z(6,n) + n + \lfloor \frac{n}{2} \rfloor + 1$	$Z(6,n) + n + \lfloor \frac{n}{2} \rfloor + 3$
40		$Z(6,n)$	$Z(6,n) + 1$	$Z(6,n) + 2$
41		$Z(6,n) + \lfloor \frac{n}{2} \rfloor$	$Z(6,n) + \lfloor \frac{n}{2} \rfloor + 1$	$Z(6,n) + \lfloor \frac{n}{2} \rfloor + 2$
42		$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor + 2$
43		$Z(6,n) + \lfloor \frac{n}{2} \rfloor$	$Z(6,n) + \lfloor \frac{n}{2} \rfloor$	$Z(6,n) + \lfloor \frac{n}{2} \rfloor + 1$
44		$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor + 2$
45		$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor + 1$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor + 2$
46		$Z(6,n) + \lfloor \frac{n}{2} \rfloor$	$Z(6,n) + \lfloor \frac{n}{2} \rfloor$	$Z(6,n) + \lfloor \frac{n}{2} \rfloor + 2$
47		$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor + 2$
48		$Z(6,n) + 4\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 4\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 4\lfloor \frac{n}{2} \rfloor + 3$
49		$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor + 2$
50		$Z(6,n) - 2n + 6\lfloor \frac{n}{2} \rfloor$	$Z(6,n) - 2n + 6\lfloor \frac{n}{2} \rfloor$	$Z(6,n) - 2n + 6\lfloor \frac{n}{2} \rfloor$
51		$Z(6,n) + 3\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 3\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 3\lfloor \frac{n}{2} \rfloor + 3$
52		$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor + 2$
53		$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor + 1$
54		$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor + 2$
55		$Z(6,n) + n + \lfloor \frac{n}{2} \rfloor$	$Z(6,n) + n + \lfloor \frac{n}{2} \rfloor + 1$	$Z(6,n) + n + \lfloor \frac{n}{2} \rfloor + 3$
56		$Z(6,n) + 3\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 3\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 3\lfloor \frac{n}{2} \rfloor + 3$
57		$Z(6,n) + 2n - 1$	$Z(6,n) + 2n - 1$	$Z(6,n) + 2n + 2$
58		$Z(6,n) + n + 3\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + n + 3\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + n + 3\lfloor \frac{n}{2} \rfloor + 4$
59		$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor + 1$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor + 3$
60		$Z(6,n) + \lfloor \frac{n}{2} \rfloor$	$Z(6,n) + \lfloor \frac{n}{2} \rfloor + 1$	$Z(6,n) + \lfloor \frac{n}{2} \rfloor + 2$
61		$Z(6,n) + n$	$Z(6,n) + n + 1$	$Z(6,n) + n + 3$
62		$Z(6,n) + 3\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 3\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 3\lfloor \frac{n}{2} \rfloor + 3$
63		$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor + 2$
64		$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor$	$Z(6,n) + 2\lfloor \frac{n}{2} \rfloor + 2$

i	G_i^6	$G_i^6 + D_n$	$G_i^6 + P_{n-1}$	$G_i^6 + C_n$
65		$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor+3$
66		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+2$
67		$Z(6,n)+n+5\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+n+5\left\lfloor\frac{n}{2}\right\rfloor+2$
68		$Z(6,n)+2n-\left\lfloor\frac{n}{2}\right\rfloor-2$	$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor+3$
69		$Z(6,n)+n+2\left\lfloor\frac{n}{2}\right\rfloor-1$	$Z(6,n)+2n-1$	$Z(6,n)$
70		$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor+3$
71		$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor+3$
72		$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor+3$
73		$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor+3$
74		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+3$
75		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+3$
76		$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor+3$
77		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+1$
78		$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor+2$
79		$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor+3$
80		$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+4\left\lfloor\frac{n}{2}\right\rfloor+4$
81		$Z(6,n)+n+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+n+3\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+n+3\left\lfloor\frac{n}{2}\right\rfloor+4$
82		$Z(6,n)+2n+\left\lfloor\frac{n}{2}\right\rfloor-1$	$Z(6,n)+2n+\left\lfloor\frac{n}{2}\right\rfloor-1$	$Z(6,n)+2n+\left\lfloor\frac{n}{2}\right\rfloor+2$
83		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+2$
84		$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor+3$
85		$Z(6,n)+n$	$Z(6,n)+n+1$	$Z(6,n)+n+3$
86		$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor+3$
87		$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor+3$
88		$Z(6,n)+n+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+n+2\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+n+2\left\lfloor\frac{n}{2}\right\rfloor+4$
89		$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor+3$
90		$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor+3$
91		$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+n+\left\lfloor\frac{n}{2}\right\rfloor+3$
92		$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+3\left\lfloor\frac{n}{2}\right\rfloor+2$
93		$Z(6,n)+2n$	$Z(6,n)+2n+2$	$Z(6,n)+n+2\left\lfloor\frac{n}{2}\right\rfloor+4$
94		$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+1$	$Z(6,n)+2\left\lfloor\frac{n}{2}\right\rfloor+3$
95		$Z(6,n)+2n-1$	$Z(6,n)+2n-1$	$Z(6,n)+2n+2$
96		$Z(6,n)+n+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+n+3\left\lfloor\frac{n}{2}\right\rfloor$	$Z(6,n)+n+3\left\lfloor\frac{n}{2}\right\rfloor+4$

i	G_i^6	$G_i^6 + D_n$	$G_i^6 + P_{n-1}$	$G_i^6 + C_n$
97		$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor + 4$
98		$Z(6,n) + 3\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 3\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 3\left\lfloor \frac{n}{2} \right\rfloor + 3$
99		$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor + 4$
100		$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor + 4$
101		$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor + 4$
102		$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor + 4$
103		$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 2$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 5$
104		$Z(6,n) + 4\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 4\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 4\left\lfloor \frac{n}{2} \right\rfloor + 4$
105		$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 5$
106		$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor + 4$
107		$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor + 4$
108		$Z(6,n) + 2n + \left\lfloor \frac{n}{2} \right\rfloor - 1$	$Z(6,n) + 2n + \left\lfloor \frac{n}{2} \right\rfloor - 1$	$Z(6,n) + 2n + \left\lfloor \frac{n}{2} \right\rfloor + 2$
109		$Z(6,n) + 2n$	$Z(6,n) + 2n + 1$	$Z(6,n) + 2n + 3$
110		$Z(6,n) + n + \left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + \left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + n + \left\lfloor \frac{n}{2} \right\rfloor + 4$
111		$Z(6,n) + n + \left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + \left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + n + \left\lfloor \frac{n}{2} \right\rfloor + 3$
112		$Z(6,n) + 2n$	$Z(6,n) + 2n + 1$	$Z(6,n) + 2n + 4$
113		$Z(6,n) + 4\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 4\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 4\left\lfloor \frac{n}{2} \right\rfloor + 3$
114		$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor + 4$
115		$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor + 4$
116		$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor + 4$
117		$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 5$
118		$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) - n + 6\left\lfloor \frac{n}{2} \right\rfloor + 5$
119		$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 2$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 6$
120		$Z(6,n) + 3\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 3\left\lfloor \frac{n}{2} \right\rfloor + 2$	$Z(6,n) + 3\left\lfloor \frac{n}{2} \right\rfloor + 4$
121		$Z(6,n) + 2n$	$Z(6,n) + 2n + 2$	$Z(6,n) + 2n + 4$
122		$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor + 4$
123		$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor + 4$
124		$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor + 2$	$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor + 5$
125		$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor + 5$
126		$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 5$

i	G_i^6	$G_i^6 + D_n$	$G_i^6 + P_{n-1}$	$G_i^6 + C_n$
127		$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + n + 3\left\lfloor \frac{n}{2} \right\rfloor + 4$
128		$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 2$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 5$
129		$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 5$
130		$Z(6,n) + 2n$	$Z(6,n) + 2n + 1$	$Z(6,n) + 2n + 4$
131		$Z(6,n) + 2n + \left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 2n + \left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 2n + \left\lfloor \frac{n}{2} \right\rfloor + 5$
132		$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 5$
133		$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor + 2$	$Z(6,n) + n + 2\left\lfloor \frac{n}{2} \right\rfloor + 4$
134		$Z(6,n) + 6\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 6\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 6\left\lfloor \frac{n}{2} \right\rfloor + 4$
135		$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 2$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 6$
136		$Z(6,n) + 2n + 4\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 2n + 4\left\lfloor \frac{n}{2} \right\rfloor + 2$	$Z(6,n) + 2n + 4\left\lfloor \frac{n}{2} \right\rfloor + 7$
137		$Z(6,n) + 2n$	$Z(6,n) + 2n + 2$	$Z(6,n) + 2n + 4$
138		$Z(6,n) + 2n + \left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 2n + \left\lfloor \frac{n}{2} \right\rfloor + 2$	$Z(6,n) - 2n + 9\left\lfloor \frac{n}{2} \right\rfloor + 7$
139		$Z(6,n) + 3n + \left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 3n + \left\lfloor \frac{n}{2} \right\rfloor + 2$	$Z(6,n) + 3n + \left\lfloor \frac{n}{2} \right\rfloor + 6$
140		$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 2$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 5$
141		$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 5$
142		$Z(6,n) + 2n + 4\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 2n + 4\left\lfloor \frac{n}{2} \right\rfloor + 2$	$Z(6,n) + 2n + 4\left\lfloor \frac{n}{2} \right\rfloor + 7$
143		$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 2$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 5$
144		$Z(6,n) + 2n + 3\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 2n + 3\left\lfloor \frac{n}{2} \right\rfloor + 3$	$Z(6,n) + 2n + 3\left\lfloor \frac{n}{2} \right\rfloor + 7$
145		$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 2n + 2\left\lfloor \frac{n}{2} \right\rfloor + 2$	$Z(6,n) + 8n - 10\left\lfloor \frac{n}{2} \right\rfloor + 6$
146		$Z(6,n) + 2n + \left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 2n + \left\lfloor \frac{n}{2} \right\rfloor + 2$	$Z(6,n) - 2n + 9\left\lfloor \frac{n}{2} \right\rfloor + 7$
147		$Z(6,n) + 2n + 3\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 2n + 3\left\lfloor \frac{n}{2} \right\rfloor + 2$	$Z(6,n) + 2n + 3\left\lfloor \frac{n}{2} \right\rfloor + 7$
148		$Z(6,n) + 3n + \left\lfloor \frac{n}{2} \right\rfloor$	$Z(6,n) + 3n + \left\lfloor \frac{n}{2} \right\rfloor + 2$	$Z(6,n) + 3n + \left\lfloor \frac{n}{2} \right\rfloor + 6$
149		$Z(6,n) + 4n + \left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 4n + \left\lfloor \frac{n}{2} \right\rfloor + 3$	$Z(6,n) + 4n + \left\lfloor \frac{n}{2} \right\rfloor + 8$
150		$Z(6,n) + 2n + 4\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 2n + 4\left\lfloor \frac{n}{2} \right\rfloor + 3$	$Z(6,n) + 2n + 4\left\lfloor \frac{n}{2} \right\rfloor + 7$
151		$Z(6,n) + 2n + 3\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 2n + 3\left\lfloor \frac{n}{2} \right\rfloor + 3$	$Z(6,n) + 2n + 3\left\lfloor \frac{n}{2} \right\rfloor + 7$
152		$Z(6,n) + 3n$	$Z(6,n) + 3n + 3$	$Z(6,n) + 3n + 6$
153		$Z(6,n) + 4n + \left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 4n + \left\lfloor \frac{n}{2} \right\rfloor + 3$	$Z(6,n) + 4n + \left\lfloor \frac{n}{2} \right\rfloor + 8$
154		$Z(6,n) + 3n + 2\left\lfloor \frac{n}{2} \right\rfloor + 1$	$Z(6,n) + 3n + 2\left\lfloor \frac{n}{2} \right\rfloor + 4$	$Z(6,n) + 3n + 2\left\lfloor \frac{n}{2} \right\rfloor + 8$
155		$Z(6,n) + 4n + 2\left\lfloor \frac{n}{2} \right\rfloor + 2$	$Z(6,n) + 4n + 2\left\lfloor \frac{n}{2} \right\rfloor + 5$	$Z(6,n) + 4n + 2\left\lfloor \frac{n}{2} \right\rfloor + 10$
156		$Z(6,n) + 6n + 3$	$Z(6,n) + 6n + 6$	$Z(6,n) + 6n + 12$

Bibliography

[1] B.M. Ábrego, S. Fernández-Merchant, G. Salazar. The rectilinear crossing number of K_n: Closing in (or are we?) In J. Pach, editor, *Thirty Essays on Geometric Graph Theory*, 563–567, Springer, 2012.

[2] B.M. Ábrego, O. Aichholzer, S. Fernández-Merchant, P. Ramos, G. Salazar. The 2-Page Crossing Number of K_n. *Discrete Comput. Geom.*, 49(4):747–777, 2013.

[3] B.M. Ábrego, O. Aichholzer, S. Fernández-Merchant, T. Hackl, J. Pammer, A. Pilz, P. Ramos, G. Salazar, B. Vogtenhuber. All Good Drawings of Small Complete Graphs. In Proc: *31st European Workshop on Computational Geometry (EuroCG)*, pp 57–60, 2015.

[4] B.M. Ábrego, O. Aichholzer, S. Fernández-Merchant, D. McQuillan, B. Mohar, P. Mutzel, P. Ramos, R.B. Richter, B. Vogtenhuber. Bishellable drawings of K_n. *SIAM J. Discrete Math.*, 32(4):2482–2492, 2018.

[5] E. Ackerman. On topological graphs with at most four crossings per edge. https://arxiv.org/abs/1509.01932, 2015.

[6] J. Adamsson, R.B. Richter. Arrangements, circular arrangements and the crossing number of $C_7 \times C_n$. *J. Comb. Theory*, Series B, 90:21–39, 2004.

[7] O. Aichholzer. On the rectilinear crossing number. Sum-

mary of results maintained at http://www.ist.tugraz.at/staff/-aichholzer/research/rp/triangulations/crossing/, last updated 2015.

[8] M. Aigner, G Ziegler. *Proofs from THE BOOK*. Springer-Verlag Berlin Heidelberg, pp. 261–268, 2010.

[9] M. Ajtai, V. Chaátal, M.M. Newborn, E. Szemerédi. Crossing-Free Subgraphs. *North-Holland Math. Studies*, 60:9–12, 1982.

[10] M. Anderson, R.B. Richter, P. Rodney. The Crossing Number of $C_6 \times C_6$. *Congressus Numerantium*, 117:97–107, 1996.

[11] M. Anderson, R.B. Richter, P. Rodney. The Crossing Number of $C_7 \times C_7$. *Congressus Numerantium*, 125:97–117, 1996.

[12] D. Archdeacon. Open Problems. In: *Topics in topological graph theory*, volume 128 of *Encyclopedia Math. Appl.*, 313–336, Cambridge Univ. Press, Cambridge, 2009.

[13] K. Asano. The crossing number of $K_{(1,3,n)}$ and $K_{(2,3,n)}$. *J. Graph Theory*, 10(1):1–8, 1986.

[14] P. Baniasadi, V.Ejov, M. Haythorpe, S. Rossomakhine. A new benchmark set for the Travelling salesman problem and Hamiltonian cycle problem. 2018. Available at https://arxiv.org/abs/1806.09285.

[15] C.A. Barefoot, R.C. Entringer. A census of maximum uniquely hamiltonian graphs. *J. Graph Theory*, 5(3):315–321, 1981.

[16] A.M.S. Barreto, H.J.C Barbosa. Graph Layout Using a Genetic Algorithm. In Proc. *Sixth Brazilian Symposium on Neural Networks*, IEEE, 179–184, 2000.

[17] C.P. Barton. The Rectilinear Crossing number for Complete Simple Graphs in E_2. PhD Thesis, The University of Texas at Austin, Austin, Texas, 1970.

[18] C. Batini, M. Talamo, R. Tamassia. Computer aided layout of entity relationship relationship diagrams. *J. Syst. Softw*, 4:163–173, 1984.

[19] L.W. Beineke, R.D. Ringeisen. On the crossing numbers of products of cycles and graphs of order four. *J. Graph Theory*, 4(2):145–155, 1980.

[20] L. Beineke, R. Wilson. The Early History of the Brick Factory Problem. *The Mathematical Intelligencer*, 32(2):41–48, 2010.

[21] D. Bokal. On the crossing number of Cartesian products with paths. *J. Comb. Th. Series B*, 97(3):381–384, 2007.

[22] D. Bokal, M. Chimani, J. Leaños. Crossing number additivity over edge cuts. *European J. of Combinatorics*, 34:1010–1018, 2013.

[23] B. Bollobas. *Modern Graph Theory.* Graduate Texts in Mathematics, Springer-Verlag New York, 1998.

[24] J. Branke, F. Bucher, and H. Schmeck. Using Genetic Algorithms for Drawing Undirected Graphs. In Proc. *Third Nordic Workshop on Genetic Algorithms and their Applications (ZNWGA)*, 193–206, 1996.

[25] G. Brinkman, J. Goedebeur, B.D. Mckay. Generation of Cubic graphs. *Disc. Math. and Theoretical Comp. Sc.*, 13(2):69–80, 2011.

[26] C. Buchheim, M. Chimani, C. Gutwenger, M. Jünger, P. Mutzel. Crossings and Planarization. *Handbook of Graph Drawing and Visualization*, Roberto Tamassia (ed.), pp 43–80, 2013.

[27] S. Cabello, B. Mohar. Crossing number and weighted crossing number of near-planar graphs. *Algorithmica*, 60(3):484–504, 2011.

[28] S. Cabello, B. Mohar. Adding One Edge To Planar Graphs Makes Crossing Number And 1-Planarity Hard. *SIAM J. Comput.*, 42:1803–1829, 2013.

[29] G. Chartrand, L. Lesniak. *Graphs and Digraphs*, Wadsworth & Brooks/Cole Advanced Books & Software, Monterey, CA, 2nd ed., 1986.

[30] M. Chimani. *Computing Crossing Numbers.* PhD Dissertation, Technical University of Dortmund, 2008.

[31] M. Chimani, C. Gutwenger. Advances in the Planarization Method: Effective Multiple Edge Insertions. *J. Graph Alg. App.*, 16(3):729–757, 2012.

[32] M. Chimani, C. Gutwenger, M. Jünger, G. Klau, W. Klein, P. Mutzel. The Open Graph Drawing Framework (OGDF). *Handbook of graph drawing and visualization*, R. Tamassia (ed.), CRC Press, 2014.

[33] M. Chimani, C. Gutwenger, P. Mutzel. Experiments on Exact Crossing Minimization Using Column Generation. *ACM J. Experimental Algorithmics*, 14(3.4):303–315, 2009.

[34] M. Chimani, C. Gutwenger, P. Mutzel, C. Wolf. Inserting a Vertex into a Planar Graph. *Proc. twentieth annual ACM-SIAM symposium on Discrete algorithms*, 375–383, 2009.

[35] M. Chimani, P. Hliněný. A tighter insertion-based approximation of the crossing number. *J. Comb. Optim.*, 33:1183–1225, 2017.

[36] M. Chimani, P. Hliněný, P. Mutzel. Vertex insertion approximates the crossing number of apex graphs. *Eur. J. of Combinatorics*, Elsevier, 33:326–335, 2012.

[37] J. Chuzhoy, Y. Makarychev, A. Sidiropolous. On graph crossing number and edge planarization. In Proc. *SODA '11 ACM-SIAM symposium on Discrete algorithms* , ACM Press, New York, 1050–1069, 2011.

[38] M. Chimani, T. Wiedera. An ILP-based Proof System for the Crossing

Number Problem. *24th European Symposium of Algorithms (ESA) 2016*, *http://crossings.uos.de*, 2016.

[39] K. Clancy, M. Haythorpe, A. Newcombe. A survey of graphs with known or bounded crossing numbers. *Submitted to Australasian Journal of Combinatorics*, submitted in 2019, Available at https://arxiv.org/abs/1901.05155.

[40] K. Clancy, M. Haythorpe, A. Newcombe. An effective crossing minimisation heuristicbased on star insertion. *J. Graph Alg. App.*, 23(2):135–166, 2019.

[41] H.S.M Coxeter. My Graph. *Proc. London Math. Soc.* 3(1):117–136, 1983.

[42] C.D.C. Cruz and D.M. Japson. The crossing number of the generalized Petersen graph $P(17, K)$, where $K \leq \lfloor \frac{17}{2} \rfloor$. *Undergraduate Thesis*, Polytechnic University of the Phillipines, 2015.

[43] A.M. Dean, R.B. Richter. The crossing number of $C_4 \times C_4$. *J. Graph Theory*, 19:125–129, 1995.

[44] E. de Klerk, D.V. Pasechnik, G. Salazar. Improved lower bounds on the book crossing numbers of complete graphs. *SIAM J. Discrete Math.*, 27(2):619–633, 2013.

[45] E. de Klerk, D.V. Pasechnik, R.B. Richter, G. Salazar. Improved Bounds for the Crossing Numbers of $K_{n,m}$ and K_n. *SIAM J. Discrete Math.*, 20(1):189–202, 2006.

[46] G. Di Battista, A. Garg, G. Liotta, R. Tamassia, E. Tassinary, F. Vargiu. An experimental comparison of four graph drawing algorithms. *Comput. Geom.*, 7(5–6), 303–325, 1997.

[47] H. Djidjev. A Linear-Time Algorithm For Finding a Maximal Planar Subgraph. *SIAM J. on Discrete Mathematics*, 20(2):444–462, 2006.

[48] E. Draženská, M. Klešč. On the crossing numbers of $G \Box C_n$ for graphs G on six vertices. *Disc. Math. Graph Th.*, 31(2):239–252, 2011.

[49] Z. Dvořák, B. Mohar. Crossing numbers of periodic graphs. *J. Graph Theory*, 83(1):34–43, 2016.

[50] P. Eades. A Heuristic for Graph Drawing. *Congressus Numerantium*, 42:149–160, 1984.

[51] R.B. Eggleton, R.K. Guy. The crossing number of the n-cube. *Amer. Math. Soc. Notices*, 17:757, 1970.

[52] T. Eloranta, E. Mäkinen. TimGA: a genetic algorithm for drawing undirected graphs. *Divulgaciones Matematicas*, 9(2):155–171, 2001.

[53] P. Erdős, R.K. Guy. Crossing Number Problems. *Amer. Math. Monthly*, 80(1):52–58, 1973.

[54] G. Exoo, F. Harary, J. Kabell. The crossing numbers of some generalized petersen graphs. *Mathematica Scandinavica*, 43:184–188, 1981.

[55] L. Faria, C.M.H. Figueiredo, O. Sýkora, I. Vrt'o. An improved upper bound on the crossing number of the hypercube. *J. Graph Theory*, 59(2):145–161, 2008.

[56] S. Fiorini. On the Crossing Number of Generilized Petersen Graphs. *North-Holland Math. Studies*, 123:225–241, 1986.

[57] S. Fiorini, J.B. Gauci. The crossing number of the generilized Petersen graph $P(3k, k)$. *Math. Bohemica*, 128(4):337–347, 2003.

[58] H. de Fraysseix, J. Pach, R. Pollack. How to draw a planar graph on a grid. *Combinatorica*, 10:41–51, 1991.

[59] J.R. Fredrickson, B. Yuan, F.C. Harris Jr. A Time-Saving Region Restriction for Calculating the Crossing Number of K_n. *Congessus Numeratium*, 168:145–158, May 2004.

[60] M.R Garey, D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness* W.H. Freeman & Co., New York, 1990.

[61] M.R. Garey, D.S. Johnson. Crossing Number is NP-Complete. *SIAM. J. on Algebraic and Discrete Methods*, 4(3):312–316, 1982/1983.

[62] L.Y. Glebsky, G. Salazar. The crossing number of $Cm \times Cn$ is as conjectured for $n \geq m(m+1)$. *J. Graph Theory*, 47(1):53–72, 2004.

[63] R. Gould. *Graph Theory.* Dover Publications, Mineola, New York, 2012.

[64] M. Grohe. Computing crossing numbers in quadratic time. *J. Comp. and System Sciences*, 68(2):285–302, 2004.

[65] C. Gutwenger. Application of SPQR-Trees in the Planarization Approach for Drawing Graphs. *PhD Dissertation*, Technical University of Dortmund, 2010.

[66] C. Gutwenger, P. Mutzel. An Experimental Study of Crossing Minimization Heuristics. *Graph Drawing 11th Int. Symposium*, LCNS, Springer, Heidelberg, 2912:13–24, 2004.

[67] C. Gutwenger, P. Mutzel, R. Weiskircher. Inserting an Edge into a Planar Graph. *Algorithmica*, Springer Science Business Media, 41:289–308, 2005.

[68] R.K. Guy. A combinatorial problem. *Bull. Malayan. Math. Soc.*, 7:68–72, 1960.

[69] R.K. Guy. Crossing numbers of graphs. In: *Graph Theory and Applications*, volume 303 Lecture Notes in Mathematics, pp. 111–124, Springer-Verlag, Berlin-Heidelberg-New York, 1972.

[70] R. K. Guy, F. Harary. On the Möbius ladders. *Canad. Math. Bull.* 10:493–496, 1967.

[71] R.K. Guy. The decline and fall of Zarankiewicz's theorem. Proof Techniques in Graph Theory, Proc. Second Ann Arbor Graph Theory Conf., Academic Press, New York, 63–69, 1968.

[72] R.K. Guy, T.A. Jenkins. The Toroidal Crossing Number of $K_{m,n}$. *J. Comb. Theory*, 6:235–250, 1969.

[73] S. Hachul, M. Junger. Drawing Large Graphs with a Potential-Field-Based Multievel Algorithm. In: Pach J. (eds) *Graph Drawing. GD 2004. Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg, 3383:285–295 , 2005.

[74] F. Harary. *Graph Theory.* Addison-Wesley, Reading, MA, 1969.

[75] F. Harary, P.C. Kainen, A.J. Schwenk. Toroidal graphs with arbitrarily high crossing numbers. *Nanta. Math.*, 6(1):58–67, 1973.

[76] H. Harborth. Parity of numbers of crossings for complete n-partite graphs. *Math. Slovaca*, 26(2):77–95, 1976.

[77] F.C Harris Jr. C.R Harris. A Proposed Algorithm for Calculating the Minimum Crossing Number of a Graph. In Proc. *Eight Quadrennial International Conference on Graph Theory, Combinatorics, Algorithms, and Applications*, Y. Alavi, A.J. Schwenk and R.L. Graham, eds., Kalamazoo, Michigan, June 1996, Western Michigan University.

[78] M. Haythorpe, A. Newcombe. On the Crossing Number of the Cartesian Product of a Sunlet Graph and a Star Graph. *B. Aust. Math. Soc.*, 100(1):5–12, 2019.

[79] P. Hliněný. Crossing number is hard for cubic graphs. *J. Comb. Theory*, Series B, 96(4):455–471, 2006.

[80] P. Hliněný, G. Salazar. On the Crossing Number of Almost Planar Graphs. *Graph Drawing 14th Int. Symposium*, LCNS, Springer, Heidelberg, 4372:162–173, 2007.

[81] P.T. Ho. On the crossing number of some complete multipartite graphs *Utilitas Mathematica*, 79:125–143, 2009.

[82] P.T. Ho. The crossing number of $K_{1,m,n}$. *Disc. Math.*, 308(24):5996–6002, 2008.

[83] J. Hopcroft, R.E. Tarjan. Efficient Planarity Testing. *J. of the ACM*, 21(4):549–568, 1974.

[84] Y. Huang, T. Zhao. The crossing number of $K_{(1,4,n)}$. *Discrete Math.*, 308(9):1634–1638, 2008.

[85] S. Jendrol, M. Ščerbová. On the crossing numbers of $Sm \times Pn$ and $Sm \times Cn$. *Casopis pro pestováni matematiky*, 107:225–230, 1982.

[86] P.C. Kainen. A Lower Bound for Crossing Numbers of Graphs with Applications to K_n, $K_{p,q}$, and $Q(d)$. *J. Comb. Theory Series B*, 12:287–298, 1972.

[87] T. Kamada, S. Kawai. An Algorithm For Drawing General Undirected Graphs. *Information Processing Letters*, 31:7–15, 1989.

[88] K. Kawarabayashi. Computing crossing number in linear time. *Proc. thirty-ninth annual ACM symposium on Theory of comp.*, 382–390, 2007.

[89] D.J. Kleitman. The crossing number of $K_{5,n}$. *J. Comb. Theory*, 9(4):315–323, 1970.

[90] M. Klešč. On the crossing numbers of cartesian products of stars and paths or cycles. *Mathematica Slovaca*, 41:113–120, 1991.

[91] M. Klešč. The crossing numbers of Cartesian products of paths with 5-vertex graphs. *Disc. Math.*, 233(1–3):353–359, 2001.

[92] M. Klešč. The crossing numbers of products of paths and stars with 4vertex graphs. *J. Graph Theory*, 18(6):605–614, 1994.

[93] M. Klešč, D. Kravecová. The crossing number of $P_n^2 \Box C_3$. *Disc. Math.*, 312:2096–2101, 2012.

[94] M. Klešč, J. Petrillová. The crossing numbers of products of paths with graphs of order six. *Disc. Math. Graph Th.*, 33(3):571–582, 2013.

[95] M. Klešč, J. Petrillová, M. Valo. On the crossing numbers of cartesian products of wheels and trees. *Disc. Math. Graph Th.*, 37(2):399-413, 2017.

[96] M. Klesc, R.B. Richter, I. Stobert. The crossing number of $C_5 \times C_n$. *J. Graph Theory*, 22(3):239-243, 1996.

[97] M. Klešč, Š. Schrötter. On the crossing numbers of Cartesian products of stars and graphs of order six. *Disc. Math. Graph Th.*, 33(3):583–597, 2013.

[98] M. Klešč, M. Valo. Minimum crossings in join of graphs with paths and cycles. *Acta Elec. Inf.*, 12(3);32–37, 2012.

[99] K. Kuratowski. Sur le problém des courbes gauches en topologie. *Fund. Math.*, 15:271–283, 1930.

[100] J. Leaños, G. Salazar. On the additivity of crossing numbers of graphs. *J. Knot Theory Ramifications*, 17:1043–1050, 2008.

[101] F.W. Levi. *Finite Geometrical Sysytems* University of Calcutta, 1942.

[102] L. Li. The Crossing Numbers of Join of a 5-Vertex Graph with Path and Cycle. *Math. Prac. Th.*, 44(11):203–2011, 2014.

[103] P.C. Liu, R.V. Geldmacher. On the deletion of nonplanar edges of a graph. *Congr. Numer.*, 24:729–738, 1979.

[104] L. Lovász, J. Pach, M. Szegedy. On Conway's Thrackle Conjecture. *Discrete and Computational Geometry*, 18(4):369–376, 1997.

[105] S.X. Lü, Y.Q. Huang. On the Crossing Numbers of $K_5 \times S_n$ *J. Mathematical Research & Exposition*, 28(3):445–459, 2008.

[106] S. Lü, Y. Huang. On the crossing number of $K_{2,4} \times S_n$. *J. Sys. Sci. Math. Sci.*, 30(7):929–935, 2010.

[107] D. Ma, H. Ren, J. Lu. Crossing Number of the Generalized Petersen Graph $G(2m+1, m)$. *J. East China Norm. Uni. (Nat. Sci. Ed.)*, 2005(1):34–39, 2005.

[108] D. McQuillan, R.B. Richter. On the Crossing Number of K_n without Computer Assistance. *J. Graph Theory*, 82(4):387–432, 2016.

[109] D. McQuillan, S. Pan, R.B. Richter. On the crossing number of K_{13}. *J. Comb. Theory*, Series B, 115:224–235, 2015.

[110] H. Mei, Y. Huang. The Crossing Number of $K_{1,5,n}$. *Int. J. Math. Combin.*, 1(1):33–44, 2007.

[111] P. Mutzel. The Crossing Number of Graphs: Theory and Computation. *LNCS 5760*, (Eds.) S. Albers, H.alt, S. Näher, 305–317, 2009.

[112] Z. Ouyang, J. Wang, Y. Huang. The crossing number of cartesian products of paths with complete graphs. *Discrete Math.*, 328:71–78, 2014.

[113] S. Pan, R.B. Richter. The crossing number Of K_{11} is 100. *J. Graph Theory*, 56:128–134, 2007.

[114] E. Jr. Pegg, G. Exoo. Crossing Number Graphs. *The Mathematica Journal*, 11(2):161–170, 2009.

[115] B. Pinontoan, R.B. Richter. Crossing numbers of sequences of Graphs I: general tiles. *Austral. J. Combin.*, 30:197–206, 2004.

[116] B. Pinontoan, R.B. Richter. Crossing numbers of sequences of Graphs II: planar tiles. *J. Graph Theory*, 42(4):332–341, 2003.

[117] C. Qian, Y. Huang. The Crossing Number of the Complete Quadruple Graph. *J. Hunan Uni. Art. Sci. Nat. Sci. Ed.*, 19(4):13–17, 2007.

[118] R.B Richter, G. Salazar. The crossing number of $C_6 \times C_n$. *Austral. J. Combin.*, 23:135–143, 2001.

[119] R.B. Richter, C. Thomassen. Intersections of curve systems and the crossing number of $C5 \times C5$. *Disc. Comput. Geom.*, 13(1):149–159, 1995.

[120] R.B. Richter, C. Thomassen. Relations Between Crossing Numbers of Complete and Complete Bipartite Graphs. *Amer. Math. Monthly*, 104(2):131–137, 1997.

[121] R.D. Ringeisen, L.W. Beineke. The crossing number of $C_3 \times C_n$. *J. Comb. Theory*, Series B, 24(2):134–136, 1978.

[122] G. Ringel. Extremel problems in the theory of graphs. In Proc: *Theory of Graphs and its Applications Symposium 1963*, 85–90, Smolenice, Publ. House Czechoslovak Acad. Sci., Prague, 1964.

[123] G. Salazar. On the crossing number of loop networks and generilized Petersen graphs. *Disc. Math.*, 302(1–3):243–253, 2005.

[124] M.L. Saražin. The crossing number of the generalized Petersen graph $P(10, 4)$ is four. *Math. Slovaca*, 47(2):189–1982, 1997.

[125] M. Schaefer. *Crossing Numbers of Graphs.* Series: Discrete Mathematics and Its Applications, CRC Press, 2017.

[126] M. Schaefer. The Graph Crossing Number and its Variants: A Survey. *Electron. J. Comb*, Third edition, DS21, Dec. 2017.

[127] N. Shanthini, J.B. Babujee. Crossing Number for $S_4 + S_N$. *Asian J. Res. Soc. Sci. Hum.*, 6(9):2030–2041, 2016.

[128] F. Sharokhi, L.A. Székely, O. Sýkora, I. Vrt'o. Drawings of Graphs on Surfaces with Few Crossings. *Algorithmica*, 16:118–131, 1996.

[129] W. Schnyder. Embedding planar graphs on the grid. In Proc. *1st ACM-SIAM Symp. on Discrete Algebra (SODA)*, San Francisco, 138–148, 1990.

[130] J. Sheehan. Graphs with exactly one hamiltonian circuit. *J. Graph Theory*, 1(1):37–43, 1977.

[131] D.A. Singer. The rectilinear crossing number of certain graphs, Manuscript, Cornell University, Ithica, New York, 1971.

[132] M. Staš. Determining Crossing Number of Join of the Discrete Graph with Two Symmetric Graphs of Order Five. *Symmetry*, 11(2):123–132, 2019.

[133] A. Steimele, W. Staton. The isomorphism classes of the generilized Petersen graphs. *Discrete Math.*, 309:231–237, 2009.

[134] Z. Su, Y. Huang. The Crossing Number of $S_5 \vee C_n$. *J. Math. Stud.*, 46(4):413–417, 2013.

[135] O. Sýkora, I. Vrt'o. On crossing numbers of hypercubes and cube connected cycles. *BIT Numerical Math.*, 33(2):232–237, 1993.

[136] U. Tadjiev, F.C. Harris Jr. Parallel computation of the minimum crossing number of a graph, In Proc. *8th SIAM Conf. on Parallel Processing for Computer Science*, 1997.

[137] P. Turán. A note of welcome. *J. Graph Theory*, 1:7–9, 1977.

[138] K. Urbanik. Solution du problème posé par P. Turán. *Colloq. Math.*, 3:200–201, 1955.

[139] I. Vrt'o. Bibliography on Crossing Numbers of Graphs. http://www.ifi.savba.sk/∼imrich, 2014.

[140] J. Wang, Y. Huang. The Crossing Numbers of Cartesian Products of Paths with 6-Vertex Graphs. *J. Jishou Uni. Nat. Sci. Ed.*, 26(2):9–13, 2005.

[141] Z. Wenping, L. Xiaohui, Y. Yuansheng, D. Chengrui. On the crossing number of $K_m \Box C_n$ and $K_{m,l} \Box P_n$. *Disc. Appl. Math.*, 156:1892–1907, 2008.

[142] W. Winterbach. *The crossing number of graphs in the plane* Master's Thesis, University of Stellenbosch, South Africa, 2005.

[143] D.R. Woodall. Cyclic-order graphs and Zarankiewicz's crossing-number conjecture. *J. Graph Theory*, 17(6):657–671, 1993.

[144] L. Xiaohui, Y. Yaungsheng, Z. Wenping, S. Lei, L. Weiming. The crossing numbers of generilized Petersen graphs of small order. *Disc. App. Math.*, 157:1016–1023, 2009.

[145] Y. Yang, G. Wang, H. Wang, Y. Zhou. The Erdős and Guy's conjectured equality on the crossing number of hypercubes. *arXiv:1201.4700v3 [math.CO]*, 2017.

[146] X. Yuan. On the crossing numbers of $K_{2,3} \vee C_n$. *J. East China Norm. Uni. Nat. Sci.*, 2011(5):21–24, 2011.

[147] K. Zarankiewicz. On a problem of P. Turan concerning graphs. *Fundamenta Mathematicae*, 41:137–145, 1954.

[148] W. Zheng, X. Lin, Y. Yang and C. Deng. On the crossing number of $K_m \Box C_n$ and $K_{m,l} \Box P_n$. Disc. Appl. Math., 156:1892–1907, 2008.

[149] T. Ziegler. *Crossing Minimization in Automatic Graph Drawing.* PhD Dissertation, Saarland University, Germany, 2001.