School of Computer Science and Software Engineering
Monash University

Bachelor of Computer Science Honours (1608), Clayton Campus

Research Proposal — Semester 1, 2005

The Analysis and Design of Approximation Algorithms for the Maximum Induced Planar Subgraph Problem

Kerri Morgan 12837008

Supervisor: Dr. G. Farr

Contents

1 Introduction

A graph is a collection of vertices (or nodes) and connections between pairs of vertices called edges. Graphs can be used to model many problems in a manner that encapsulates the essential elements of the problem.

Road networks can be modelled by graphs. Locations can be represented as vertices and roads connecting locations as edges. Global Positioning Systems (GPS) used by emergency service vehicles may in some cases use such a model of the road network in order to navigate to locations in the most efficient manner possible. Software engineering diagrams use graphs to model complex problems in a simple way whilst retaining essential information. For example, in structure charts functions are represented as vertices and edges represent function calls. Graphs can also be used to model social networks. In this case each vertex represents an individual (or group of individuals) and the edges represent connections between individuals (or groups of individuals). A family tree is a simple example of a graph of this type. In this case the nodes represent individuals and the edges represent family connections such as the parent-child relationship and the husband-wife relationship. Graphs are frequently used to model electronic circuits where the edges represent wires and the vertices represent connections between the wires.

One famous problem that was solved using graph theory concerned the seven bridges in the city of Königsberg. Was it possible to take a journey that traversed all seven bridges but did not cross any bridge more than once? In 1735, Leonhard Euler proved that it was not possible. Euler modelled the problem as a graph where vertices represented land masses, and each of the edges represented a bridge connecting two land masses. He then showed that a graph contains such a journey if and only if every vertex has even degree. Euler's paper, *Solutio problematis ad geometriam situs pertinentis*, on the problem was published in 1736 and is considered to be the first paper on graph theory.

When modelling real-world problems, it is important to be able to produce a good two-dimensional representation of a graph. Such a representation is used to display the graph on a screen or print it on a page. Circuit design also uses a two-dimensional representation of the graph with the layout of the wires on the circuit board being represented by the layout of the graph.

However, a two-dimensional representation of the graph is not always sufficient. Many applications require than no edges cross. A graph that can be drawn in the plane without edge crossings is a planar graph. Many applications take advantage of this representation and other properties of planarity. For example, in graph drawing the layout of a graph so that no edges cross is used in circuit design. In this case, a two dimensional representation with edges crossing would be unsatisfactory as this would represent wires crossing which would result in short circuits.

While planarity is a useful property, many graphs are nonplanar. Planarisation is the process of finding a portion (preferably the largest) of the graph that is planar. Many problems are easier to solve when restricted to the class of planar graphs. Algorithms for solving problems on planar graphs may in some cases be able to be used on nonplanar graphs by first planarising the graph. A graph can be planarised either by the removal of vertices (and their incident edges) or by the removal of edges.

The Maximum Induced Planar subgraph problem is the task of removing the smallest possible number of vertices from a graph to produce a planar subgraph. Unfortunately this problem is known to be NP-hard [9]. Approximation algorithms provide a possibly suboptimal solution in a reasonable amount of time.

The aim of this project is to design and analyse new approximation algorithms for the Maximum Induced Planar Subgraph problem. The behaviour of the algorithms will be compared with some of the existing approximation algorithms in terms of running time and performance.

In this document, the context of the Maximum Induced Planar Subgraph problem in graph theory is examined. A general plan for the research method into designing new algorithms for this problem, and for analysing the behaviour of these new algorithms and existing algorithms is then presented. A brief review of the work that has been completed for this project is provided including a section detailing current modifications and improvements. Finally, the proposed timeline and deliverables for this project are listed.

2 Definitions

This section introduces some standard definitions that will occur throughout this document.

Subgraph A graph $G^{'} = (V^{'}, E^{'})$ is a subgraph of $G = (V, E)$ if $V^{'} \subseteq V$ and $E^{'} \subseteq E$

Induced Subgraph A graph $G^{'} = (V^{'}, E^{'})$ is an induced subgraph of $G = (V, E)$ if $V^{'} \subseteq V$ and $E' = \{uv | u, v \in V^{'} \wedge uv \in E\}$.

Hereditary A property is said to be hereditary if for every graph G with the property, every subgraph $H \subseteq G$ also has the property.

Planarise To find a planar subgraph of a graph.

Approximation Algorithm An algorithm for an optimisation problem that produces a feasible but not necessarily optimal solution in a reasonable amount of time.

Performance Ratio The ratio of the size of solution produced in comparison to the size of the optimal solution. In the context of the planarisation of graphs, let c be the number of vertices in the planar graph produced by an approximation algorithm. Let k be the number of vertices in the optimal solution, that is the largest planar subgraph. Then the performance ratio of the algorithm on this graph is c/k. The performance ratio of an algorithm for a class of graphs, provides a lower bound of the performance ratios of the algorithm on all graphs in the class. The algorithm will do at least as well as this ratio on every graph in the class.

3 Research Context

3.1 Planarity

A planar graph is a graph that can be drawn in the plane so that no two edges cross. Consider a graph represented in this way on a plane. If the sections of the plane on which the vertices and edges were drawn were removed, the remaining areas would correspond to connected regions called faces. Euler's polyhedron theorem [14, p.9], commonly known as Euler's formula, states that any connected planar graph with n vertices, m edges, and f faces satisfies $n - m + f = 2$. This formula has a more general form namely $f - m + n - c = 1$ which applies for all planar graphs of c components.

From this theorem it can be shown that any planar graph with $n \geq 3$ vertices has at most $3n - 6$ edges. If the planar graph is also bipartite, it has at most $2n - 4$ edges.

These inequalities are useful in proving that certain graphs cannot be drawn in the plane. For example, K_5 (see Figure 1) is not planar as it has 5 vertices and 10 edges, which does not satisfy the inequality $m \leq 3n - 6$ as $10 \not\leq 9$. The bipartite graph $K_{3,3}$ (see Figure 2) is also not planar as it has 6 vertices and 9 edges which does not satisfy the inequality $m \leq 2n - 4$ as $9 \not\leq 8$.

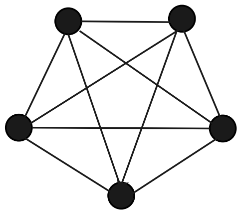

Figure 1: The graph K_5.

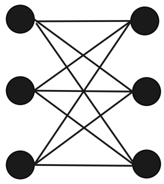

Figure 2: The graph $K_{3,3}$.

Kuratowki showed a graph is planar if and only if it does not contain a subgraph homeomorphic to either K_5 or $K_{3,3}$ [8]. The graphs G_1 and G_2 are homeomorphic if they can both be obtained from a graph G by a series of subdivisions. A subdivision of an edge uv of G adds a new vertex w to $V(G)$ and replaces the edge uv with two edges uw and wv.

Planarity is a hereditary property of graphs, so that if a graph G is planar then any subgraph H of G is also planar. This hereditary property of planar graphs allows the use of many "divide-and-conquer" algorithms on problems involving planar graphs [12].

3.2 Planar Separator Theorem

Consider the task of dividing a planar graph into two disjoint planar subgraphs by removing a subset of the vertices. It is sometimes preferable that each of these two subgraphs contains a significant proportion of the vertices of the original graph, and that the number of vertices removed from the original graph is as small as possible. A planar separator theorem shows that no more than some small proportion of vertices is required to be removed from any planar graph in order to produce two substantial disjoint planar subgraphs.

Lipton and Tarjan [11] showed that the vertices of any n-vertex planar graph "can be partitioned into three sets A, B, C such that no edge joins a vertex in A with a vertex in B, neither A nor B contains more than $2n/3$ vertices, and C contains no more than $2\sqrt{2n}$ vertices". As A and B have no edges in common, removing the vertices in C from the graph separates the graph into two subgraphs. Due to the hereditary nature of planarity, both these subgraphs are also planar. Lipton and Tarjan's paper provides an algorithm which finds a partition satisfying the above properties in $O(n)$ time. Hence, Lipton and Tarjan provided the means to effectively divide a planar graph into two planar subgraph, each of a reasonable size, by removing only a small number of vertices. The two subgraphs are planar and thus can also be divided by the same process.

Many algorithms have combined Lipton and Tarjan's separator theorem with divide and conquer techniques on planar graphs to provide more efficient solutions for problems such as boolean circuit complexity, graph embedding, pebbling, and maximum matching [12].

However, many graphs are nonplanar. It may be possible in some cases to apply these algorithms to nonplanar graphs, by first planarising the graph. A solution for the planar subgraph can then be found by using the "divide-and-conquer" techniques previously discussed. Let R denote the vertices removed to form the planar subgraph. These vertices also form an induced subgraph. This graph can be planarised and the techniques for planar subgraphs be applied to its planar subgraph. By repetitively applying this technique until no vertices remain in the set R, a set of solutions can be found for a series of planar subgraphs of the original graphs. None of these subgraphs have any vertices in common, but there may be edges in the original graph that connect two subgraphs. If the solutions for the subgraphs can be effectively combined, a solution for the original graph can be found.

Algorithms for drawing nonplanar graphs used in applications such as circuit layout also often initially planarise the graph, then utilise one of the many existing algorithms for drawing planar graphs [7].

3.3 Planarisation

It has been seen that it is useful to be able to find a planar subgraph of a graph. A subgraph of a graph can be formed either by removing edges or removing vertices from the original graph.

The Maximum Planar Subgraph problem for a graph $G = (V, E)$ is the task of finding the largest set $E^{'} \subseteq E$ such that the subgraph $G^{'} = (V^{'}, E^{'})$ where $V^{'} = \{v \in V \wedge v$ is incident to $e \in E^{'}\}$ is planar. A similar problem is the Maximal Planar Subgraph problem where the subgraph produced may not be of maximum size, but no edge can be added to the set of edges in the subgraph without destroying the planarity property.

The Maximum Planar Subgraph problem is known to be NP-hard [13]. This necessitates the use of good approximation algorithms in order to find a solution in a reasonable amount of time. The solutions found may not be optimal.

Several approximation algorithms have been studied for the Maximum Planar Subgraph Problem. Călinescu, Fernandes, Finkler and Karloff [2] produced an approximation algorithm for finding planar subgraphs with a performance ratio of 4/9.

The Maximum Induced Planar Subgraph problem for a graph $G = (V, E)$ is the task of finding the largest set $V' \subseteq V$ such that the subgraph $G' = (V', E')$ where $E' = \{uv | uv \in E \wedge u, v \in V'\}$ is planar. Again there is a related Maximal Induced Planar Subgraph problem. In contrast to the Maximum Planar Subgraph problem, this problem has had little attention. The Maximum Induced Planar Subgraph problem is known to be NP-hard [9]. Thus good approximation algorithms are required to find a solution that may not be optimal in a reasonable amount of time.

Halldórsson and Lau [6] produced a linear time algorithm for graphs of maximum degree at most d with a performance ratio of $1/\lceil(d+1)/3\rceil$. They achieve this by partitioning the graph into at most $1/\lfloor(d+1)/3\rfloor$ subgraphs of degree at most 2. Each of these parts consists of a forest and is thus planar. The largest of these parts is chosen as the planar subgraph. This algorithm is limited to producing planar subgraphs of maximum degree at most 2.

Edwards and Farr [3] presented a polynomial time algorithm for graphs of n vertices and maximum degree at most d. This algorithm finds an induced planar subgraph of at least $3n/(d+1)$ vertices, which implies a performance ratio of $3/(d+1)$. This is at least comparable to the Halldórsson and Lau algorithm, but in cases where $d \not\equiv 2 \pmod 3$ is considerably better. The subgraphs produced by Edwards and Farr's algorithm are not constrained to having maximum degree of at most two as are those produced by Halldórsson and Lau's algorithm. The algorithm divides the vertices of the graph into two sets, R (those not in the induced planar subgraph) and P (those in the induced planar subgraph). P is initially empty. The algorithm creates a planar subgraph by incrementally adding vertices from the original graph to P or interchanging a vertex in P with one in R whilst maintaining planarity of the graph $\langle P \rangle$. The restrictions on the selection of vertices to be added to P are stricter than required to maintain planarity, but enable certain properties in the graph to be maintained which allows the algorithm's performance to be analysed.

The subgraph produced by this algorithm is not necessarily maximal. The authors note that in some cases it is possible to add an additional vertex to P after the algorithm has stopped whilst maintaining planarity.

Another algorithm for finding large induced planar subgraphs by Edwards and Farr [5] achieves a similar bound on performance to their previous algorithm, but can be shown to guarantee the same performance on additional classes of graphs, namely graphs of average degree at most d. That is, for a graph of n vertices with average degree at most $d \geq 4$, or a graph that is connected and has average degree $d \geq 2$, it finds an induced planar subgraph of at least $3n/(d+1)$ vertices. As in their previous algorithm, this algorithm again divides the set of vertices in the original graph into two sets P and R. However, in this case P initially contains all vertices of the original graph. $r(G)$ is defined to be the reduced graph of G, where G is reduced by a series of operations that can be applied whilst maintaining planarity. A graph is considered to be reduced when none of these operations can any longer be applied. The algorithm iteratively removes the vertex of highest degree in the graph $r(\langle P \rangle)$ whilst $\langle P \rangle$ is nonplanar. For any graph G, let $p(G)$ be the size of the smallest set X of vertices of G such that $G - X$ is planar. It can be shown that $p(G) \leq \sum_{v \in V(r(G))}((d_{r(G)}(v) - 2)/(d_{r(G)}(v) + 1))$ [5]. To avoid the cost of testing for planarity each iteration, Edwards and Farr use a loop condition that the size of R must be less than $\rho = \left\lfloor \sum_{v \in V(r(G))}((d_{r(G)}(v) - 2)/(d_{r(G)}(v) + 1)) \right\rfloor$.

In this article the former algorithm by Edwards and Farr will be referred to as the Vertex Addition algorithm, whilst the latter algorithm will be referred to as the Vertex Removal algorithm. Both these algorithms give scope for further research. Each guarantees an induced planar subgraph of at least some bounded number of vertices. By examining the behaviour of the algorithms in practice on randomly generated graphs, more information may be gleaned regarding the performance bounds in terms of the size of the subgraphs found. The performance bounds of the Halldórsson and Lau algorithm are the same as that of the Vertex Removal algorithm when $d \equiv 2 \pmod 3$. A comparison of the actual performance of these two algorithms in practice would give a measure of the additional benefit that can be derived by not limiting the subgraph to having maximum degree at most 2.

Forest subgraphs are inherently planar. The planar subgraphs produced by Halldórsson and Lau's algorithm are of degree at most two. Components of these subgraphs are either paths or cycles. Some of the power of the first algorithm by Edwards and Farr is gained by operations that encourage subgraphs that maintain either a tree-like structure or outerplanarity. However, it does not favour operations that maintain these properties above the other operations. Liebers comments in her discussion on the Maximal Induced Planar Subgraph problem that she is unaware of any work investigating the impact of different vertex orderings [10]. It is possible that the order in which the vertices are added to P in the Vertex Removal algorithm may affect the size of the subgraph produced by the algorithm. By favouring vertices whose addition does not destroy the tree structure (and when this is no longer possible outerplanarity), the planar subgraph remains in a state which maximizes the possibilities for including additional vertices.

3.4 Fragmentability

Consider the task of removing as few vertices as possible from a graph in order to break the graph into small components. One method to achieve this is to first use the Maximum

Induced Planar Subgraph problem. The algorithms by Edwards and Farr, and Halldórsson and Lau, provide a bound on the minimum proportion of vertices required to be removed in order to planarise a nonplanar graph. A planar graph can then be divided into two planar subgraphs by removing relatively few vertices using an algorithm such as the one presented by Lipton and Tarjan in their paper on the planar separator theorem [11]. This process can be repetitively performed on the planar subgraphs until the entire graph has been broken into small components. Only a small number of vertices are required to be removed on each iteration. Together the total number of vertices required for all of these iterations and the number of vertices removed in order to planarise the original graph provide the number of vertices whose removal will break the graph into components of at most some size C. It is interesting to consider how few vertices can be removed in order to fragment a graph into components of at most size C.

Fragmentability provides a measure of how vulnerable graphs are to being broken into components of bounded size by the removal of a small number of vertices. A graph $G = (V, E)$ is (C, ϵ)-fragmentable if and only if there exists a set $X \subseteq V$, $|X| \leq \epsilon|V|$, whose removal results in $G \setminus X$ consisting of components of size at most C. A class of graphs is said to be ϵ-fragmentable if and only if there exists an integer C such that all graphs in the class are (C, ϵ)-fragmentable. The coefficient of fragmentability for a class of graphs Γ, denoted by $c_f(\Gamma)$, is defined as $c_f(\Gamma) = \inf\{\epsilon : \Gamma \text{ is } \epsilon\text{-fragmentable}\}$.

Edwards and Farr [4] considered the fragmentability of classes of graphs. Any class of graphs for which there exists an appropriate separator theorem is known to have $c_f(\Gamma) = 0$. A separator theorem exists for planar graphs [11]. Hence for any $0 \leq \epsilon \leq 1$ there exists an integer C such that any planar graph of n vertices can be broken into components of a most size C by removing at most ϵn vertices.

It was shown that if one class of graphs could be reduced to another class of graphs by removing a proportion of the vertices, and the coefficient of fragmentability is known for the second class, then an upper bound on the coefficient of fragmentability for the first class could be calculated. Edwards and Farr proved that given two classes of graphs Γ and $\Gamma^{'}$ that if for any graph $G \in \Gamma$ by removing at most $\epsilon|V(G)| + A$ vertices (where ϵ is a non-negative real number and A is a non-negative integer) the resulting graph was an element of $\Gamma^{'}$, then $c_f(\Gamma) \leq c_f(\Gamma^{'}) + \epsilon - \epsilon c_f(\Gamma^{'})$.

The Maximum Induced Planar Subgraph problem calculates the minimum number of vertices required to remove to produce an induced planar subgraph. Edwards and Farr used this to calculate an upper bound on the proportion of vertices that must be removed from graphs belonging to certain classes of graphs to produce a planar subgraph. For example, graphs of maximum degree at most d and n vertices were shown to be able to be made planar by removing at most $(d-2)n/(d+1)$ vertices. As previously mentioned planar graphs have $c_f(\Gamma) = 0$. If for a class of graphs Γ, any graph can be reduced to a planar subgraph by removing at most ϵn vertices, then the coefficient of fragmentability for this class is at most ϵ. Thus if an algorithm for finding the Maximum Induced Planar subgraph can be shown to find a planar subgraph for any graph belonging to a class Γ by removing at most ϵ vertices, then the coefficient of fragmentability for this class of graphs is at most ϵ. It is noted that an approximation algorithm may remove more vertices compared to an algorithm that finds the optimal solution. If the approximation algorithm for the task requires at most ϵ vertices to be removed, whereas finding the optimal solution requires at most ϵ' vertices to be removed, then $c_f(\Gamma) \leq \epsilon' \leq \epsilon$, so the upper bound of $c_f \leq \epsilon$ still holds although it may not be a tight upper bound.

Edwards and Farr's analysis of their Vertex Addition and Vertex Removal algorithms has provided an upper bound on the number of vertices required to be removed by these algorithms to planarise graphs of maximum degree at most d and of average degree at most d respectively [3, 5]. These bounds provide upper bounds for the coefficient of fragmentability for these classes of graphs. Thus, analysis of algorithms for the Maximum Induced Planar Subgraph problem can provide useful information about the fragmentabililty of a class of graphs.

Conversely, the coefficient of fragmentability can be used to provide a lower bound on the proportion of vertices to remove for certain classes of graphs in the Maximum Induced Planar Subgraph problem. For example, Edwards and Farr were able to show that the class of graphs, Γ_d with maximum degree at most d and $d \geq 2$, has $(d-2)/(2d-2) \leq c_f(\Gamma_d) \leq (d-2)/(d+1)$. If for any graph in Γ_d, the removal of at most ϵn vertices will produce a planar subgraph, then $c_f(\Gamma_d) \leq \epsilon$. Combining these results an lower bound for ϵ can be found, namely $(d-2)/(2d-2) \leq \epsilon$.

4 Research Plan and Methods

4.1 Research Methods

In this project the Halldórsson-Lau, Vertex Removal and Vertex Addition algorithms will be implemented. Currently a Vertex Subset Removal algorithm is being worked on by Keith Edwards and Graham Farr. It is intended if possible to also implement this algorithm. The behaviour in terms of the size of subgraph produced and the running time of these algorithms will be examined by running these programs on a series of graphs.

Graphs will be randomly generated using a number of random graph generators, including the Steger-Wormald [15] algorithm for generating random d-regular graphs, the Bollobás [1] algorithm for generating random d-regular graphs and Erdös' classical graph generator for graphs of average degree d. The Bollobás method generates using a uniform distribution on d-regular graphs. However, it is much slower than the Steger-Wormald method. The Steger-Wormald method uses a probability distribution that is similar but not identical to Bollobás' method, but as it is much faster the size of graphs it can produce within a reasonable time is larger. However, as the Steger-Wormald generator proved too slow when creating very large graphs, an alternative generator (the Morgan generator) was developed by the author that produces large d-regular graphs quickly. Results of the approximation algorithms on graphs produced by the various graph generating algorithms will be compared. If the results are similar on smaller graphs, it may be possible to use the results on larger graphs produced by the Morgan generator to draw conclusions on the behaviour of the various algorithms on very large graphs.

The next task in the project is to use the results about the behaviour of the existing algorithms to aid in the design and implementation of a hybrid algorithm. This algorithm will be based on the Vertex Addition and Vertex Removal algorithms. At this stage the algorithm will make a decision for each vertex whether to assign it to P, the planar set, or R, the removed set. All vertices initially belong to an unassigned set U.

Next a new approximation algorithm for the Maximum Induced Planar Subgraph problem will be designed. If possible, some mathematical analysis of this algorithm will be performed. As mentioned earlier there is some scope to examine the effects of ordering the selection of vertices to be included in the planar subgraph. It may be possible to include a heuristic function that favours vertices whose addition maintains outerplanarity. In the Vertex Addition algorithm Edwards and Farr have an operation that interchanges vertices. They note that this operation maintains the size of P but decreases the number of edges in $\langle P \rangle$. However, in addition it decreases the number of cycles in $\langle P \rangle$ by at least one. It may be possible to design an algorithm that initially creates a planar subgraph that is a forest, then adds vertices whose addition creates as few cycles as possible in the resulting subgraph. By introducing multiple cycles in a component as late as possible in the algorithm, outerplanarity is maintained as long as possible.

Finally, the behaviour of all algorithms will be examined by running them on a series of randomly generated graphs. The algorithms will be compared on the basis of running time and size of the planar subgraph produced. Both the worst case and average case will be considered.

4.2 Current Project Status

4.2.1 Implementation of Algorithms

Both the Vertex Addition and Vertex Removal algorithms have been implemented. These programs have been tested on a number of small graphs and in each case have successfully produced the required planar subgraph.

Three graph generators have been implemented. Initially, a generator was implemented based on the Steger-Wormald algorithm [15], but was found to be too slow when generating 20-regular graphs of $10,000$ vertices. A second generator was implemented based on a modified version of the Steger-Wormald generator. However, this also proved too slow for large graphs. Finally, an algorithm was designed by the author. This algorithm has yet to be fully analysed, but it would appear to have a similar probability distribution to the Steger-Wormald distribution. A generator based on this algorithm produces a 20-regular graph of $10,000$ vertices in approximately sixty seconds. The use of this generator will allow the performance of the approximation algorithms to be tested on much larger graphs than could be produced by the other generators in a reasonable amount of time.

Currently, a function is being implemented, that allows information about the planar subgraph produced by the various approximation algorithms to be saved. Each planar subgraph is checked to see that the number of vertices and the number of edges satisfies Euler's inequality $m \leq 3n - 6$. While a graph that satisfies this inequality may not be planar, a graph that does not satisfy this inequality cannot be planar. Currently all subgraphs produced by the approximation algorithms implemented have satisfied this inequality. The behaviour of the algorithms on randomly generated graphs of up to $10,000$ vertices has been tested.

4.2.2 Improvements and modifications

In the Vertex Removal algorithm, Edwards and Farr [5] use the loop condition that the size of R must be less than $\rho = \left\lfloor \sum_{v \in V(r(G))} ((d_{r(G)}(v) - 2)/(d_{r(G)}(v) + 1)) \right\rfloor$. This avoids the cost of testing for planarity each iteration The value of ρ provides the maximum number of vertices required to be removed from P. However, it is not a sufficient condition for the loop. For many graphs, the reduced graph becomes the null graph while $|R| < \rho$. In these cases the loop condition holds, but the instructions within the loop cannot be executed, as there are no vertices in the reduced graph to remove. The condition required for the algorithm to perform correctly was ($|R| < \rho$ and $|r(\langle P \rangle)| \neq 0$). Thus, the Vertex Removal algorithm was implemented with this modification.

In their discussion on the Vertex Addition algorithm Edwards and Farr understate the power of the step of type (ii) when they claim that "any step of type (ii) or (iii) decrease $|E(P) + k(P)|$ by exactly 1" [3]. In fact any step of type (ii) decreases $|E(P) + k(P)|$ by at least 1. Step (ii) occurs when a vertex x_i is selected in P to be interchanged with a vertex x_0 in R. The vertex selected lies on a circuit formed by edges in $\langle P \rangle$ and edges from vertices in P incident to x_0. It is known by the selection criteria that vertex x_i has degree at least 3 in $\langle P \rangle$. The only situation when the value $|E(P) + k(P)|$ is exactly one is when the degree in $\langle P \rangle$ of the vertex x_i being removed from P is three. The increased benefit of this action may prove useful in the development of future algorithms.

4.3 Proposed Thesis Chapter Headings

1. Introduction
   (a) Background
   (b) Aims
2. Planar Graphs
   (a) Background
   (b) Planar Separator Theorem
   (c) Fragmentability of Planar Graphs
   (d) The Usefulness of Planarity
3. Planarisation of graphs
   (a) Background
   (b) Maximum Planar Subgraph Problem
   (c) Maximum Induced Planar Subgraph Problem
   (d) Approximation Algorithms for the Maximum Induced Planar Subgraph Problem
      i. Existing Algorithms
      ii. Vertex Subset Removal Algorithm
      iii. Hybrid Algorithm
      iv. New Algorithm
4. Random Graph Generation
   (a) Erdös' Classic Graph Generator
   (b) Bollobás Random Graph Generator
   (c) Steger-Wormald Random Graph Generator
   (d) Morgan Graph Generator
5. Experiments
   (a) Description of Tests
   (b) Results
6. Discussion of Results
7. Conclusion and Further Investigation
8. Bibliography

Appendix A Results

Appendix B Programs

4.4 Timetable

Table 1: Projected Timetable

Task	Completion Date
Implement Vertex Addition Algorithm [3]	23rd March
Implement Quick Random Graph Generator [15]	1st April
Implement Vertex Removal Algorithm [5]	8th April
Implement modified version of Quick Random Graph Generator [15]	15th April
Research Proposal	27th April
Implement a program for collection of data on behaviour of algorithms	3rd May
Implement Erdös' Classical Graph Generator	10th May
Design and implement random graph generator	12th May
Collect some data on behaviour of algorithms currently implemented	13th May
Interim Presentation	2nd June
Literature Review Draft	10th June
Implement Hybrid Algorithm	17th June
Implement Halldórsson and Lau [6]	8th July
Literature Review	27th July
Implement Vertex Subset Removal Method (if available)	5th August
Testing performance of the above algorithms	12th August
Development of own approximation algorithm	31st August
Thesis Draft	7th September
Further testing of performance of all algorithms	21st September
Final Presentation	24–28th October
Submit Thesis	1st November
Submit Log Book	1st November
Complete Web Site	10th November

4.5 Special Facilities Required

The facilities offered Honours students at Monash University (Clayton) are sufficient to do the research.

5 Deliverables

1. Design a hybrid algorithm based on Vertex Addition and Vertex Removal algorithms.
2. Design a new approximation algorithm for the Maximum Induced Planar Subgraph problem.
3. Implement existing approximation algorithms for the Maximum Induced Planar Subgraph problem.
   (a) Vertex Removal algorithm
   (b) Vertex Addition algorithm
   (c) Halldórsson and Lau algorithm
   (d) Vertex-subset Removal algorithm (if available)

4. Implement algorithms designed by author for the Maximum Induced Planar Subgraph problem
   (a) Hybrid algorithm
   (b) New algorithm
5. Implement random graph generators
   (a) Bollobás method
   (b) Steger-Wormald method
   (c) Erdös method
   (d) Morgan method
6. Experimental Study
   - Investigate the behaviour of the above approximation algorithms on a series of randomly generated d-regular graphs and a series of randomly generated graphs of average degree d. Data will be collected on the results of these algorithms for both average-case and worst-case in the following areas:
     - Performance in terms of size of subgraph produced.
     - Running time.
   - Analyse and compare the behaviour of the above approximation algorithms based on the data collected
7. Final Thesis

References

[1] B. Bollobás. A probabilistic proof of an asymptotic formula for the number of labelled regular graphs. *European Journal of Combinatorics*, 1:311–316, 1980.

[2] G. Călinescu, C. G. Fernandes, U. Finkler, and H. Karloff. A better approximation algorithm for finding planar graphs. *Journal of Algorithms*, 27:269–302, 1998.

[3] K. Edwards and G. Farr. An algorithm for finding large induced planar subgraphs. In P. Mutzel, M. Jünger, and S. Leipert, editors, *Graph Drawing: 9th International Symposium, GD 2001*, Lecture Notes in Computer Science 2265, pages 75–83. Springer-Verlag, Berlin, 2001.

[4] K. Edwards and G. Farr. Fragmentability of graphs. *Journal of Combinatorial Theory (Series B)*, 82:30–37, 2001.

[5] K. Edwards and G. Farr. Planarization and fragmentability of some classes of graphs. Technical Report 144, School of Computer Science and Software Engineering, Monash University, 2003.

[6] M. M. Halldórsson and H. C. Lau. Low-degree graph partitioning via local search with applications to constraint satisfaction, max cut,and colouring. *Journal of Graph Algorithms and Applications*, 1:1–13, 1997.

[7] M. Jünger and P. Mutzel. Maximum planar subgraphs and nice embeddings: practical layout tools. *Algorithmica*, 16:33–59, 1996.

[8] C. Kuratowski. Sur les problèmes des courbes gauches en Topologie. *Fundamenta Mathematicae*, 15:271–283, 1930.

[9] J. M. Lewis. The node-deletion problem for hereditary properties is NP-complete. *Journal of Computer and System Sciences*, 20:219–230, 1980.

[10] A. Liebers. Planarizing graphs — a survey and annotated bibliography. *Journal of Graph Algorithms and Applications*, 5:1–74, 2001.

[11] R. J. Lipton and R. E. Tarjan. A separator theorem for planar graphs. *SIAM Journal on Applied Mathematics*, 36:177–189, 1979.

[12] Richard J. Lipton and Robert Endre Tarjan. Applications of a planar separator theorem. *SIAM Journal on Computing*, 9:615–627, 1980.

[13] P.C. Liu and R.C. Geldmacher. On the deletion of nonplanar edges of a graph, Proceedings of the Tenth Southeastern Conference on Combinatorics, Graph Theory and Computing, 1977. *Congressus Numerantium*, 24:727–738, 1979.

[14] T. Nishizeki and N. Chiba. *Planar Graphs : Theory and Algorithms*. Annals of Discrete Mathematics 32. Elseveir Science, Amsterdam, 1988.

[15] A. Steger and N.C. Wormald. Generating random regular graphs quickly. *Combinatorics, Probability and Computing*, 8:377–396, 1999.