An investigation into Chromatic Graph Theory and optimal Graph Colouring techniques

Developing and implementing a graph coloring tool and analysing it's performance on a range of generic and special-case graphs

Harry Alexander Richards under the supervision of Professor Daniel Paulusma

A literature review contributing towards a bachelors degree in
Computer Science and Mathematics within the Natural Sciences programme
Bachelor of Science School of Engineering and Computing Sciences

Durham University
28/09/2018

Glossary of key terms

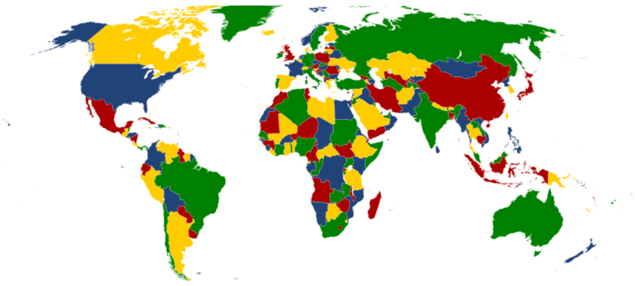
Adjacent: Two vertices are adjacent if there exists an edge connecting them.
Chromatic number: The chromatic number of a graph is the smallest number of colours needed to colour the vertices of the graph such that no two adjacent vertices share the same colour. $\chi(G)$ is the chromatic number of G.
Colouring: A graph coloring is a labelling of the vertices of a graph by elements from a given set of colors.
First-fit/Greedy algorithm: The first-fit/greedy algorithm iterates through each of the vertices and assigns the first colour that satisfies the requirements of the GCP to the vertice. It is simply concerned with colouring a graph and not the number of colours it uses.
Graph: The fundamental object of study in graph theory, a system of vertices connected in pairs by edges.
Graph Colouring Problem (GCP): The graph colouring problem is an assignment of colours to vertices of a given graph such that each vertice is coloured differently to the vertices adjacent to it.
Heuristic: A heuristic, is any approach to problem solving that employs a practical method, not guaranteed to be optimal, perfect, logical, or rational, but instead sufficient for reaching an immediate goal.
Hypergraph: A hypergraph is a generalisation of a graph in which an edge can join any number of vertices.
Metaheuristic: A metaheuristic is a higher-level procedure or heuristic designed to find, generate, or select a heuristic (partial search algorithm) that may provide a sufficiently good solution to an optimization problem.
NP-complete (non-deterministic polynomial time): An NP-complete problem can be verified quickly (in polynomial time), however there is no known efficient way to locate a solution in the first place; the most notable characteristic of NP-complete problems is that no fast solution to them is known.
Planar graph: A graph that can be drawn in such a way that no edges cross.
Travelling Salesman Problem (TSP): The TSP is a mathematical problem in which one has to find which is the shortest route which passes through each of a set of points once and only once.

When speaking about ‘colouring the graph’ and other equivalent phrases, we are referring to colouring the vertices of the graph - independent of colouring the edges of the graph (unless stated otherwise).

Abstract

In graph theory, the graph colouring problem is an assignment of colours to vertices of a given graph such that each vertice is coloured differently to the vertices adjacent to it. Graph colouring is of interest for its applications, it deals with the fundamental problem of partitioning a set of objections into classes, according to certain rules and there are many problems of this nature. Many deep and interesting results have been obtained during the hundreds of years that graph colouring has been studied, despite this there are very many easily formulated, interesting problems left - throughout the course of this project my intention is to identify and investigate a number of these.

Introduction and Background



A depiction of the Four Colour Theorem applied to a simple map.

Graph colouring has its origins in the work of Francis Guthrie who, while studying at University College London in 1852, noticed that no more than four colours ever seemed necessary to colour a map while ensuring that neighbouring regions received different colours. This observation captured the interests of Augustus De Morgan, William Hamilton, Arthur Cayley, Charles Pierce, and Alfred Kempe all of whom were unable to provide a conclusive proof for the conjecture. It eventually took more than 120 years, and the considerable use of large-scale computing resources to prove conclusively what is now know as the Four Colour Theorem. To achieve this proof it was identified that any map can be converted into a corresponding planar graph. As noted above a graph is simply an object comprising vertices (nodes), some of which being linked by edges (lines); a planar graph is a special type of graph that can be drawn such that none of the edges cross. We produce a planar graph by drawing vertices to represent regions and connect vertices with edges if the corresponding regions share a border with one another. Once we have produced our planar graph, our task has now been converted to colour the vertices appropriately (such that adjacent vertices are not the same colour). Map-colouring dominated the field for many years but over time the theory became more general, abastract and applicable. There are many such problems that can be converted into corresponding graph colouring problems, and by colouring the graph appropriately we can find a solution. However, as with many real-life problems some solutions are better than others, and an optimal solution often corresponds to the colouring of the graph that satisfies the requirement of two adjacent vertices not using the same colour but also uses the least number of colours. This minimal number of colours used is referred to as the chromatic number and $\chi(G)$ is the notation for the chromatic number of the graph G. From the four-colour problem, the theory has developed into a many-sided body of problems, theories, results and applications, and despite the number of problems that have been solved, there are an abundance of simply stated but intrinsically challenging problems and the number of these is increasing with time.

Chromatic graph theory deals with the fundamental process of partitioning a set of objects into classes in accordance with certain rules, in chromatic graph theory these rules are elementary as for each pair of objects it is unambiguous whether they may exist together in the same class. Finding the chromatic number of both generic and special-case graphs will be the theme of this project, despite the simplicity of the rules this does not indicate the problems encountered are simple - on the contrary. Thus as you may expect chromatic graph theory is a popular area of study and it attracts many active researchers. The task of computing the chromatic number of a graph is an NP-complete problem which has contributed to the attraction of this area of mathematics - in fact, determining where a graph is 3-colourable is itself NP-complete.

Problem Themes and Environment

The following are considered some of the most striking results in chromatic graph theory in relatively recent years:

- the 5-list colourability of planar graphs (dating back ot V. G. Vizing inn 1975 and to P.Erdos, A. L. Rubin and H. Taylor in 1979) by Thomasen

- the confirmation by Robertson, Sanders, Seymour and Thomas of the truth of the four-colour theorem (F. Guthrie and A. De Morgan (1852))
- the asymptotic solution by Reed of the problem as to whether for $k \geq 9$ there are k-chromatic graphs without complete k-graphs and of maximum degree k (V. G. Vizing (1968) and O. V. Borodin and A. V. Kostochka (1977))
- the proiof by Chudnovsky, Robertson, Seymour and Thomas of the perfect graph conjecture of C. Berge around 1960
- the proof by Thomassen of the weak 3-flow conjevture of W. T. Tutte (1953)
- the solution by Kostochka and Yancey to the problem of critical graphs with few edges (due to T. Gallai (1963) and O. Ore (1967))
- the description found by Borodin, Dvorak, Kostochka, Lidicky and Yancey of all 4-critcial planar graphs with exactly four triangles (B. Grunbaum (1963), V. A. Aksenov (1974) and P. Erdos (1990)).

Along with these major achievments there have been a multitude of suprising solutions presented in chromatic graph theory in recent times, as we see in 'Graph Coloring Problems' [**?**] there are an abundance of easily stated problems in the field. As more solutions are acquired the area of study continues to surprise and if we were to consider which of the original 211 problems in the book would be solved first, we would be faced with a stark contrast to the reality.

Identitied Themes

There are many interesting and challenging issues within chromatic graph theory such as the impressive feats presented above, and when choosing an area of study within it we are spoilt for choice in terms of resources and popular projects in the current age. However, as alluded to in this document my project will focus on utilising structural data regarding graphs and how we can utilise this to acquire near optimal upper bounds for the chromatic number of given graphs - through the utilisation of both heuristics and algorithms. Developing heuristics and algorithms which are competent when acquiring upper bounds for the chromatic number of graphs with given structural data will be of great difficulty and require much research, on top of this within the department there are a number of special-case graphs which will provide the project with additional focus.

Proposed Project Direction

The proposed direction of the project may be illustrated best by what is expected of the graph colouring tool that I'm developing. These objectives range in complexity from basic which I expect to complete in the near future, to more complicated objectives and innovation which can hopefully make a real contribution to the field and give us an insight into developing faster algorithms that produce optimal colourings when given suitable heuristics surrounding the nature of the graphs structure.

Trivially it is necessary for us to be able to input given graphs into the tool and whilst this should be completed in the near future it is fundamental in order to be able to apply any algorithm I develop to a graph. There are specific benchmark graphs and classifications of graphs that we will want to input initially, however as the project develops I expect that progressively more complex special-case graphs will become more of a focus. Thus, I should design the input tool in a way that is as general as possible, limiting the number of amendments I will have to make in the future. It is also necessary that any graph stored is stored efficiently, in order to do this I will need to research efficient graph representations that can be easily manipulated and iterated through. This will be of even greater importance should it be necessary or of interest to be capable of storing and colouring mixed hypergraphs in the future.

Once I am able to store graphs efficiently I can begin to implement basic algorithms such as the first-fit algorithm in order to solve the graph colouring problem. However, when doing this it is not expected that the solutions yielded will be optimal and in many cases not even clase to optimal. Each solution will be recorded and we will alter factors that affect the efficacy of the algorithm. In the example of the first-fit algorithm we will alter the order in which vertices are coloured and analyse the effect this has on the solution. Analysing the results should make it clear that the structure of the graph plays a significant role in the suitability of an algorithm for graph colouring.

At this point it should be clear that the success to be expected from basic algorithms such as first-fit is limited and that in order to be able to perform optimal or near-optimal colourings more information is required. In order to improve the results yielded, and approach the smallest known upper bounds for chromatic numbers for the benchmark graphs (referenced below) I will research into heuristics and how I'm able to develop practical heuristics for graphs of a specific structure. Through developing and implementing these heuristics and reducing the number of colours required to solve the GCP we should then gain a greater understanding of key properties which affect the chromatic number $\chi(G)$ of a graph G. This will be a key focus throughout the project as well as other algorithmic considerations when performing graph colouring.

In order to verify that any results yielded by any graph-colouring tool developed throughout the project we will need a graph colouring checker. At this point we should have implemented an efficient method of storing coloured graphs and utilising the stored data we should be able to verify that the colouring yielded satisfies the requirements of the GCP - that is that no two adjacent vertices are assigned the same colour.

Graph colouring optimisation

In terms of improving the algorithms that are initially created there are many possibilities we can explore. Initially, I will develop some evolutionary algothm (perhaps with the features of a genetic algorithm) as I've had success with algorithms following the same principles in the past when finding optimal tours for the travelling salesman problem. I will also go on to develop an ant-based/ant colony algorithm. Should I have time I will implement a simulated annealing algorithm, after which I will compare the effectiveness of each of the strategies implemented i.e. which produced the best upper bound for the chromatic number of each graph, and I will present an analysis of key factors in graph structure which alter the effectiveness of an algorithm. There is a considerable amount of existing research for the simulated annealing algorithm as well as recorded results, I will utilise these to ensure the academic reasoning behind my analysis is sound.

Evolutionary algorithms

I will implement an evolutionary algorithm and currently this is likely to be a genetic algorithm (prior to further research). The genetic algorithm is inspired by the process of natural selection and is a member of the larger class of evolutionary algorithms.

Ant-based/Ant colony algorithm

The ant-based or ant colony optimisation algorithm is a probabilistic technique for solving computational problems which can essentially be reduced to finding good paths through graphs. Artificial ants represent multi-agent methods inspired by the behaviour of ants in real life.

Simulated annealing algorithm

Simulated annealing is a probabilistic technique for approximating the global optimum of a given function, specifically in a large search space for an optimisation problem such as in the GCP. The term annealing comes form annealing in metallurgy, which is the process/technique from which the simulated annealing algorithm is inspired.

Combinations of the above techniques (i.e. artifical ants) and local search algorithms can be utilised for optimisation tasks such as the Graph Colouring Problem, and I intend to explore the effectiveness in my project.

Implementation

From the papers that I have looked over it seems that the majority of graph colouring algorithms implemented which utilise ant-based or simulated annealing algorithms are implemented in `C++`, I plan on looking into this and potentially developing my own algorithms in `C++` - however if I do not see any substantial benefit I expect I will implement them in Python due to my familiarity with the language.

Conclusions

A practical and hopefully effective route for the project is to produce my own tool in order to solve the Graph Colouring problem, and to some extent this will be simple (if we only expect a result) as we can simply utilise simple known algorithms such as the first-fit algorithm. In order to build on this is essential that I decide upon an efficient method of representing/storing what I'm sure will eventually be very sophisticated special-case graphs. From here I can begin to develop metaheuristics based around evolutionary algorithms, ant-colony algorithms and simulated annealing and I hope that these will prove effective in terms of the upper bounds they yield for the chromatic number. Once these have been implemented I hope to work on an hybrid algorithm such as a evolutionary algorithm (e.g. genetic algorithm) that utilises ant-colony algorithm principles as this is something I have explored in the TSP with great results. I also believe that this will provide the project with a unique direction and ensure that there is little overlap with existing projects within this area. Once completed the steps are not set in stone however I hope to have performed much more research as well as experimentation regarding the effectiveness of particular algorithms on graphs of certain structure. At this point I shall hopefully be able to meander from the general trends of which algorithms perform well on graphs which structure and focus in on developing algorithms for special-case graphs.
This is merely an outline for what I expect of the project and following guidance I expect this may be altered somewhat.

Bibliography

[1] A Guide to Graph Colouring
Book
Available at: http://opencarts.org/sachlaptrinh/pdf/16446.pdf
R.M.R. Lewis, 2016.

[2] Graph Colouring: An Ancient Problem with Modern Applications
PDF Document

Available at: https://doi.org/10.1080/2058802X.2016.11963998
Rhyd Lewis, 2016.

[3] Topics in chromatic graph theory
Book
Available at: https://doi.org/10.1017/CBO9781139519793
Lowell W. Beineke and Robin J. Wilson with academic consultant Bjarne Toft, 2015.

[4] Graph coloring problems
Book
Available at: https://doi.org/10.1002/9781118032497.fmatter
Tommy R. Jensen and Bjarne Toft, 1995.

[5] Efficient Graph Representations
Book
Available at: https://doi.org/10.1090/fim/019
Jeremy P. Spinrad, 2003.

[6] Graph Colouring
e-PDF
Available at: http://www-sop.inria.fr/members/Frederic.Havet/Cours/coloration.pdf
F. Havet, 2004.

[7] Six Degrees: The Science of a Connected Age
Book
Available at:
Duncan J. Watts, 2003.

[8] Six Degrees: The Science of a Connected Age (Columbia University summary)
Powerpoint
Available at: https://icr.ethz.ch/taicon/events/watts/slides.pdf
Columbia University, Date Unknown.

[9] DIMACS Graphs: Benchmark Instances and Best Upper Bounds
Webpage
Available at: http://www.info.univ-angers.fr/pub/porumbel/graphs/
Daniel Porumbel, Jin Kao Hao and Pascale Kuntz , 2008.

[10] Graph Coloring Instances
Webpage
Available at: https://mat.gsia.cmu.edu/COLOR/instances.html
David Johnson, Joe Culberson, Gary Lewandowski, Craig Morgenstern and Michael Trick, Date Unknown.

[11] Glossary of graph theory terms
Webpage
Available at: https://en.wikipedia.org/wiki/Glossary_of_graph_theory_terms
Wikipedia, 2018.

[12] Graph coloring
Webpage

Available at: https://en.wikipedia.org/wiki/Graph_coloring
Wikipedia, 2018.

[13] Graph colouring problem based on discrete imperialist competitive algorithm
e-PDF
Available at: https://doi.org/10.5121/ijfcst.2013.3401
Hojjat Emami and Shahriar Lotfi, 2013.

[14] Graph colouring problems and their applications in scheduling
e-PDF
Available at:
Daniel Marx, 2003.

[15] Graph Representations and Algorithms
Powerpoint
Available at: https://web.stanford.edu/class/archive/cs/cs106b/cs106b.1126/lectures/24/Slides24.pdf
Daniel Marx, 2003.

[16] Graph theory with applications
Book
Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.721.3161&rep=rep1&type=pdf
J. A. Bondy and U. S. R. Murty, 1976.

[17] Applications of Graph Coloring
e-PDF
Available at: https://link.springer.com/chapter/10.1007/11424857_55
Unal Ufuktepe and Goksen Bacak, 2005.

[18] Graph colouring and applications
e-PDF
Available at: http://www-sop.inria.fr/members/Frederic.Havet/habilitation/intro.pdf
F. Havet, 2004.

[19] Chromatic Graph Theory
Book
Available at: http://web.xidian.edu.cn/zhangxin/files/20150825_221833.pdf
Gary Chartrand and Ping Zhang, 2009.

[20] Topics in Graph Colouring and Graph Structures
Thesis
Available at: http://personal.lse.ac.uk/fergusod/thesis.pdf
David G. Ferguson, 2013.

[21] A Graph Coloring Algorithm for Large Scheduling Problems
e-PDF
Available at: https://nvlpubs.nist.gov/nistpubs/jres/84/jresv84n6p489_a1b.pdf
Frank Thomson Leighton, 1979.

[22] Hybrid Evolutionary Algorithms for Graph Coloring
e-PDF
Available at: https://link.springer.com/article/10.1023/A:1009823419804
Philippe Galinier and Jin-Kao Hao, 1999.

[23] Optimization by Simulated Annealing: An Experimental Evaluation; Part II, Graph Coloring and Number Partitioning
e-PDF
Available at: https://pubsonline.informs.org/doi/abs/10.1287/opre.39.3.378
David S. Johnson, Cecilia R. Aragon, Lyle A. McGeoch and Catherine Schevon, 1991.

[24] Some experiments with simulated annealing for coloring graphs
e-PDF
Available at: https://www.sciencedirect.com/science/article/abs/pii/S0377221787801480
M.Chams, A.Hertz and D.de Werra, 1987.

[25] Probabilistic bounds and heuristic algorithms for coloring large random graphs
e-PDF
Available at: https://s2.smu.edu/~matula/Tech-Report82.pdf
Abhai Johri and David W. Matula, 1982.

[26] Representing graphs
Article
Available at: https://www.khanacademy.org/computing/computer-science/algorithms/graph-representation/a/representing-graphs
Abhai Johri and David W. Matula, 1982.

[27] Ant colony optimization algorithms
Webpage
Available at: https://en.wikipedia.org/wiki/Ant_colony_optimization_algorithms
Wikipedia, 2018.

[28] Genetic algorithm
Webpage
Available at: https://en.wikipedia.org/wiki/Genetic_algorithm
Wikipedia, 2018.

[29] Simulated annealing
Webpage
Available at: https://en.wikipedia.org/wiki/Simulated_annealing
Wikipedia, 2018.