A new algorithm for the Maximum Independent Set Problem

Lachlan Ashcroft
Griffith University

Gold Coast, Australia
lochieashcroft@griffithuni.edu.au

Abstract—**This paper contains a brand new algorithm for identifying maximum independent sets in a graph as well as its complement. A literature review was performed to give context on what the current research is in this domain and as a point of comparison for the effectiveness of the algorithm.**
Index Terms—**independent, set, MIS, algorithm**

I. Introduction

Given an unweighted, undirected graph $G = (V, E)$ where V is set of vertices and E a set of edges, an independent set contains vertices in which no two vertices are adjacent. A maximal independent set is a set in which the inclusion of any additional vertices would introduce a connecting edge into the set. The maximum independent set (MIS) of a graph is the largest maximal independent set. The maximum size of the MIS is N, where N is the number of vertices however this is only if G is a completely disconnected graph. The MIS is a NP hard optimization problem [1].

MIS is closely related to the maximum clique problem, by definition a clique in Gc (complement of G) is an independent set in G, by this logic the maximum independent set can be found by solving the maximum clique problem in the complement of in the input graph.

There is also a relationship between MIS and the minimum vertex cover problem, however the algorithm presented in this paper focuses purely on the MIS.

The maximum independent set of a graph has various real world applications and because of this an efficient algorithm is highly desired. This paper is structured as follows, section 2 contains the literature review, section 3 the XYZ algorithm, section 4 the results and lastly the conclusion is section 5.

Tests will be ran against a subset of the DIMACS[2] in order to benchmark XYZ algorithm.

A. *Terminology*

The degree of a vertex v shown as $d(v)$ is the number of edges (connections) of vertex v.

A neighbourhood is all vertices directly connected to an individual vertex.

The degree of the neighbourhood $dn(v)$ is the sum of the degrees of all vertices in the neighbourhood.

Adjacency matrix is a matrix of dimensions (N, N) where N is the number of vertices, a value of 1 indicates two vertices are adjacent.

Adjacency list/set contains all the vertices which are connected to an individual vertex.

II. Literature Review

A literature review was performed in order to understand the current state of the research regarding the MIS, used as inspiration for the development of a new algorithm, and lastly these algorithms can be used to assess the effectiveness of the developed algorithm.

As mentioned in the introduction the MIS is related to both the maximum clique and minimum vertex cover problems. During the research it was found that many approaches to solve the MIS is to simply translate the problem into one of these other problems, find a solution and convert back. This is a valid solution for the MIS and should still be included in the literature review to some degree, even though the developed algorithm explicitly solves the MIS.

[3] Michael Luby converted Monte Carlo style algorithms into deterministic algorithms with the same parallel running time in 1986. The expected running time of the entire algorithm is $O((logn)^2)$ using $O(m)$ processors. This alone shows that parallel randomized algorithms may be an efficient approach for solving the MIS. Another parallelised algorithm proposed at the University of California at Berkley [4] has an expected running time of $O((logn)^4)$ using $O(n^3/(logn)^3)$ processors, the randomized version of the algorithm is $O((logn)^3)$ with $O(n^2)$ processors.

Both of these algorithms are of order $O(logn)$ but to achieve this run time a large number of processors are required, which not feasible for a significant sized of n which is the number of vertices in the graph.

In 1995 a greedy randomized adaptive search algorithm was proposed [5]. In the first phase it iteratively builds a set with a random greedy approach, the heuristic values for the vertices are then updated every iteration. This algorithm can also be ran in parallel which increases the effectiveness. The testing went up to 8 processors and the results are very promising, especially considering if this were to be ran on modern hardware.

An algorithm states it has a run time of $1.1966^n n^{(}O(1))$ and $1.1893^n n^{(}O(1))$ for graphs with a maximum degree of 6 and $1.1970^n n^{(}O(1))$ for 7 [6]. It does this by making a careful analysis on the structure of these bounded degree

graphs. These expected run times are very good compared to others however they are very limited in their application. An extension could be to split a graph into multiple sub graphs in which each sub graph would of equal vertex degrees, use this sort of algorithm to the find the MIS, and then combine sets to find the MIS of the entire graph.

As stated before the MIS can be solved by finding the maximum clique in the input graphs complement, in 2013 a new algorithm was developed [7]. A single threaded algorithm was developed which is a branch and bound derivative that happend to out perform existing algorithms. This algorithm was then parallelised distributing the search tree between multiple CPU cores, using 12 CPU cores that algorithm provided up to 2 orders of magnitude faster execution on large benchmarks.

An algorithm designed with a run time complexity of $O(n^4)$ has been developed [8]. The algorithm is greedy based and is recursive. It picks the vertex with the highest degree, all vertices not connected are put into a set, if collisions are present they the vertex of the highest degree is removed, and this process is repeated. A graph is included showing the relationship between the time taken to find the MIS and the graph density with 1000 vertices, there is a direct correlation between a lower run time and high densities.

III. Algorithm

The algorithm takes inspiration from ......

A. Design

Just some ideas which may be explored

B. Parallel Greedy

Sort the vertices by their degree in ascending order, randomly pick a starting vertex from the first 10% of nodes. Randomly pick another vertex to add to the set while maintaining the independence with a bias for a vertex with a low degree. This is performed on multiple threads.

C. Random shrink and grow

Randomly choose N vertices to build a set, if the set is independent randomly add 1 or more vertices to the set. If the set is not independent randomly remove vertices until it is.

D. Greedy neighbourhood degree

Similar to previous ideas however instead of using the degree of a node for the greedy algorithm use the neighbourhood degree.

E. Multi neighbourhoods

Find multiple maximal independent sets or almost maximal and combine them to a much larger independent set, the idea is that the resulting set will be somewhat close to the maximum Independent set. To combine sets vertices may need to be removed.

F. Pseudocode

Developed algorithm's pseudocode will be here ...

IV. Results

Develoepd algorithm DIMACS benchmarks, compared to task sheet results and algorithms in the literature review ...

References

[1] Garey. M. R, Johnson. D. S: Computers and Intractability: A Guide to the theory NP – completeness. San Francisco: Freeman (1979).

[2] DIMACS clique benchmarks. Benchmark instances made available by electronic transfer at dimacs.rutgers.edu, Rutgers Univ., Piscataway. NJ. (1993).

[3] Luby. M: A simple parallel algorithm for the maximal independent set problem: - SIAM journal on computing (1986)

[4] Richard M. Karp, Avi Wigderson: A Fast Parallel Algorithm for the Maximal Independent Set Problem: - Journal of the Association for Computing Machinery. Vol 32, No. 4 1985

[5] Thomas A. Feo, Mauricio G. C. Resende, Stuart H. Smith: A Greedy Randomized Adaptive Search Procedure for Maximum Independent Set: - Operations Research Vol. 42, No. 5 1994

[6] Mingyu Xiao, Hiroshi Nagamochi: Exact Algorithms for Maximum Independent Set: - Information and Computation, 08/2017, Volume 255

[7] Matjaž Depolli†, Janez Konc, Kati Rozman, Roman Trobec, and Dušanka Janežič: Exact Parallel Maximum Clique Algorithm for General and Protein Graphs - J. Chem. Inf. Model., 2013, 53 (9), pp 2217–2228

[8] Ahmad Abdel-Aziz Sharieh, Mohammed H. Mahafzah, Ayman Al Dahamsheh: An Algorithm for Finding Maximum Independent Set in a Graph: - European Journal of Scientific Research ISSN 1450-216X Vol.23 No.4 (2008), pp.586-596