Approximating Minimum Vertex Cover

Zaymon Foulds-Cook s5017391
School of Information Communication Technology
Griffith University, Gold Coast Campus
Gold Coast, QLD, Australia

Abstract—**This paper defines a real coded approach to approximating the minimum vertex cover of dense graphs...**

I. INTRODUCTION

A. Problem Statement

The minimum vertex cover (mvc) given by:

$$\text{mvc} = V' \subset V \text{ in graph } G(V,E)$$
$$\text{such that } uv \in E \Rightarrow u \in V' \vee v \in V'$$

That is to say the minimal set of vertices where every edge in graph $G(V,E)$ has at least one endpoint to a vertex in the vertex cover. Developing efficient algorithms for the mvc problem has applications in various fields. The SNP assembly problem in computation biochemistry can be resolved by finding the minimum vertex cover in order to resolve conflicts between sequences in a sameple (?, ?). Another use for finding the mvc is in computer networking security as a team of computer scientists affiliated with the 'Virology and Cryptology Lab' and the French Navy, 'ESCANSIC' "... have recently used the vertex cover algorithm [6] to simulate the propagation of stealth worms on large computer networks and design optimal strategies for protecting the network against such virus attacks in real-time." (?, ?).

The vertex cover problem can be applied to the traveling salesman problem. Vertex cover was also used as the main algorithmic engine in a method of nearest-neighbor data classification applicable in intelligent systems and data analysis (?, ?).

The mvc problem is classified in the NP-hard class of problems and was proven to be NP-Complete in a landmark paper in 1972 (?, ?). The mvc problem is a classical optimization problem in the field of computer science and finite combinatorics. The minimum vertex cover can also be represented as:

$$\text{all } V \notin \text{Maximum Independent Set of}$$
$$compliment(G(V,E))$$

II. LITERATURE REVIEW

A multitude of approximation algorithms exist for finding approximate vertex covers in various graphs. One such algorithm is the Djikstra based algorithm proposed by Chen (?, ?). An advantage of this algorithm is the bounding exponential time complexity of $\mathcal{O}(n^3)$ where n is the number of vertices in the graph. A disadvantage to the approach is the approximation ratio is 2. This approximation while fast is not effective at finding vertex covers close to optimality.

Another approach which has been investigated is a local search method with greedy heuristics. An approach proposed by Balaji uses heuristics to generate quality solutions for a variety of graphs (?, ?). The approach when compared experimentally with hBOA and Dual-LP algorithms produces solutions closer to optimality, especially diverging from the comparison algorithms as the size of the graph increases. The algorithm uses guiding heuristics in a greedy manner to determine which vertex to add to the expanding vertex cover. Another greedy heuristic driven approach proposed by Tomar, uses the maximum degree of vertices in the current evaluated neighborhood as the guiding heuristic (?, ?).

An approach proposed by Pullan leverages a new maximum clique algorithm called Phased Local Search (?, ?) and applies it to the maximum independent set/minimum vertex cover problem. The algorithm applies a series of sub-algorithms alternating between phases of incremental growth and plateau searching. The incremental growth component will rapidly increase the size of the maximum independent set or vertex cover until there are no more candidates to be added to the set. Then a phase of plateau search uses different methods and operators to swap and evaluate different combinations of nodes with countermeasures to avoid stagnation caused by cycling or nodes which confuse the initial heuristic (?, ?).

III. ALGORITHM DESCRIPTION

IV. ALGORITHM PERFORMANCE

V. RESULTS

VI. CONCLUSION