Arab J Sci Eng (2014) 39:7003–7011
DOI 10.1007/s13369-014-1159-7

RESEARCH ARTICLE - COMPUTER ENGINEERING AND COMPUTER SCIENCE

Maximum Independent Set Approximation Based on Bellman-Ford Algorithm

Mostafa H. Dahshan

Received: 20 May 2013 / Accepted: 31 October 2013 / Published online: 31 May 2014

Abstract This paper presents a new algorithm for approximating the solution of the maximum independent set (MIS) problem. The proposed algorithm takes a novel approach of treating the MIS problem as a least-cost path problem, using an adapted version of Bellman–Ford algorithm, in which all vertices are used as sources, and the cost of a vertex is measured as the number of vertices excluded if this vertex is included in the independent set. Analysis of the proposed algorithm shows that its running time is approximately $O(n(n^2 - m))$, where n is the number of vertices and m is the number of edges. Extensive tests against several benchmark graphs and random-generated graphs show a significant improvement of the proposed algorithm over other approximation algorithms, including one of the best known ones such as Greedy algorithm.

Keywords Maximum independent set · Combinatorial problems · Approximation algorithms · Graph algorithms

الخلاصة

تعرض هذه الورقة خوارزمية جديدة لتقريب حل مشكلة أكبر مجموعة مستقلة (MIS). إن الخوارزمية المقترحة تأخذ نهجا جديداً لعلاج مشكلة MIS كمشكلة المسار الأقل تكلفة ، وذلك باستخدام نسخة مكيفة من خوارزمية بيلمان-فورد. وتستخدم فيها جميع النقاط بوصفها نقاط بداية ويتم قياس تكلفة النقطة بعدد النقاط المستبعدة إذا تم تضمين هذه النقطة في المجموعة المستقلة. إن تحليل الخوارزمية المقترحة يُظهر أن وقت تشغيلها هو تقريبا $O(n(n^2 - m))$ ، حيث n هو عدد النقاط و m هو عدد الوصلات. وتظهرُ الاختبارات الواسعة مع العديد من الرسوم المعدة للقياس والرسوم المولدة عشوائياً تفوقاً كبيراً للخوارزمية المقترحة على الخوارزميات التقريبية الأخرى ، بما في ذلك واحدة من أفضل الخوارزميات المعروفة منها مثل خوارزمية الجشع.

M. H. Dahshan (✉)
Department of Computer Engineering, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia
e-mail: mdahshan@ksu.edu.sa

1 Introduction

The maximum independent set (MIS) problem, also known as the stable set problem, is a fundamental discrete optimization problem. Given an undirected graph $G = (V, E)$, where V is a set of vertices and E is a set of edges, an independent set $I_G \subseteq V$ is a set of vertices that are not adjacent, i.e., $\forall u, v \in I_G, (u, v) \notin E$. The cardinality of I_G is called the independence number $\alpha(G)$. The MIS problem is to find such a set I_G with maximum $\alpha(G)$. This problem is known to be NP-hard [1] and is also hard to approximate [2]. When using graphs to model real-life situations where a vertex represents an object and an edge represents a conflict between two objects, MIS can be used to find the largest set of objects with no conflicts. As such, the MIS problem has many applications in various areas. Examples include label placement [3], map labeling [4], packet scheduling [5], wireless link scheduling [6], Bluetooth scatternet formation [7], and multi-object tracking [8], to name a few.

The goal of this paper is to approximate the solution of the MIS problem using a systematic rather than an empirical approach. We try to approximate the NP-hard problem by a polynomial-time one, then use a proven-optimal algorithm for the polynomial-time problem to solve the approximated one. To our knowledge, this approach has not been used before in any approximation algorithm for the MIS problem.

The rest of the paper is organized as follows: Sect. 2 provides a literature review on related works. Section 3 provides a detailed description of the proposed algorithm, with an illustrative example and an analysis of the algorithm complexity. Section 4 shows the results of extensive tests performed on the proposed algorithm using several benchmark and random-generated graphs. Finally, Sect. 5 summarizes the findings of the paper and gives an insight on possible future improvements.

2 Related Works

The research on the MIS problem is mainly focused on three main areas. The first area is the development of exact solutions with lower running times. A recent survey study [9] shows that the best known exact solution for the worst-case complexity MIS has a running time in $O(2^{n/4})$, where n is the number of vertices. Some research works on exact solutions focus on graphs with bounded degree. For example, an exact algorithm is presented in [10] with $O^*(1.0977^n)$ running time for graphs with a maximum degree of 3. Another exact algorithm is presented in [11] with $O^*(1.1737^n)$ running time for graphs with degree bounded by 5.

The second area is obtaining a lower bound on the independence number $\alpha(G)$. One of the best known lower bounds on the independence number is $\alpha(G) \geq \sum_{v \in V} \frac{1}{d_G(v)+1}$, where $d_G(v)$ is the degree of vertex v [12].

The third main area is the development and evaluation of approximation algorithms. In this area, Greedy and Greedy-like algorithms have gained most researchers' interests, and several proposed approximation algorithms were built on them [13]. Greedy algorithm (also known as Greedy Minimum or Greedy Min) selects the vertex u with minimum degree $d_G(u)$, deletes it with its neighbors from G and repeats the process until G is empty. The independent set is composed of the selected vertices. Greedy is one of the best known approximation algorithms, as discussed in [14]. It can find an independent set of size $(1+o(1))n \ln(d)/d$ for sparse graphs, and size $(1+o(1)) \log_2 n$ for random graphs, where $d = 2m/n$ is the average degree of a graph G with n vertices and m edges [15].

In [16], the authors define a set of Greedy Max-type algorithms, which select a vertex u from G according to a selection rule, delete u together with its connected edges, and repeat the process on the resulting graph until it contains no more edges. The independent set contains the remaining vertices in the graph. The Greedy Max-types are determined by the criteria of the selection rules. For example, criterion C1 selects u with maximum degree among its neighbors. Criterion C2 (classical Greedy Max) selects u with maximum degree in G. The Vertex Support Algorithm (VSA) proposed in [17] is similar to Greedy Max, but it selects the vertex u with maximum support, where support is defined as the sum of the degrees of u's neighbor vertices.

In [18], the authors describe a greedy sequential algorithm for MIS that loops over vertices in an arbitrary order, adding a vertex to the MIS only if none of its neighbors has already been added. The authors show that this algorithm is highly parallel. They present an approach of using prefixes to balance between parallelism and redundancy, and describe linear work parallel algorithms based on this approach.

In [19], the author presents a heuristic algorithm for finding MIS by converting the graph into a finite partially ordered set, which is partitioned into minimum number of chains and used to create a special directed graph on which the algorithm is applied. The algorithm is tested against some benchmark graphs. The running time of the algorithm is of high degree polynomial $O(n^8)$. However, the author notes that the theoretical importance of the algorithm is the probability of constructing a polynomial-time solution for some NP-complete problems.

In this paper, we are interested in the area of development of approximation algorithms. Because Greedy algorithm has received such intensive analysis and evaluation, we use it as a baseline algorithm to compare with. In addition, we also consider VSA and two of the Greedy Max-type algorithms.

The main problem with Greedy and similar algorithms discussed above is that once a vertex is selected to be added to the independent set at any iteration, it cannot be replaced if it appeared to be a bad choice at subsequent iterations. A motivation of our work is to have the algorithm keep track of vertices selected over multiple iterations. This allows finding multiple possible MIS instances and choosing the best instance among them.

3 Proposed Algorithm

The proposed algorithm, Bellman–Ford based MIS Algorithm (BMA), redefines MIS as a problem of finding the least-cost path, by measuring the cost of a vertex u as the number of vertices that are excluded if u is included in the path. It uses an adapted version of Bellman–Ford algorithm which uses all vertices as sources, and thus can find at most $n = |V|$ different paths. In this context, each path represents an independent set of vertices.

The goal of BMA algorithm is to minimize the number of excluded vertices for each vertex added to the path. The pseudo-code for BMA algorithm is shown in Algorithm 1. BMA algorithm accepts a graph $G = (V, E)$ as an input and outputs a set I_G which is the approximated MIS. The algorithm works on multiple rounds. To keep track of all found paths, the algorithm maintains the lists *Exclude*, *Cost* and *Previous* for each round. BMA algorithm can run for up to $|V| - 1$ rounds. Each new round is a new attempt to find a higher independence number. The algorithm will stop at the round where no more vertices can be added to any path (independent set). Although BMA algorithm returns only one independent set, it actually keeps track of all found independent sets, even non-optimal ones. The algorithm can easily be modified to return multiple independent sets with maximum size, if such sets exist, without any additional computational cost. In what follows, we provide a detailed description of BMA algorithm and further explain it by an example.