A FASTER APPROXIMATION ALGORITHM FOR INDEPENDENT SET PROBLEM

KRISHNA KUMAR SINGH, LAKKARAJUGOVINDA

Abstract: This paper proposes a polynomial-time of order $O(n^4)$ algorithm for finding maximal independent set in a graph. For any graph with n vertices and maximum degree of vertex Δ must have a maximum independent set of size at least $\lceil n/(\Delta+1)\rceil$. An improved lower bound $(n+\Delta-\delta)/\Delta$ and upper bound $(n+2-\delta)/2$ of independent set are obtainedinthispaper. The algorithm finds a maximum independent set for most of the known graphs. A set of vertices are chosen on the basis of minimum degree of vertex and the maximum number of edges removed toward the remaining graph. The performance ratio obtained is far better than the performance ratio obtained so far, when Δ, and δ are comparable and close to n.
Keywords: Independent set, Heuristics, performance bound

Introduction: A maximal independent set is an independent set of all non-adjacent vertices of a graph. A maximum independent setis a largest independent set for a given graph *G* and its size is denoted by $|I(G)|$. The problem of finding such a set is called the maximumindependent set problem and is an NP-hardoptimization problem. To deal with intractability some heuristics are used to find a better approximate solution. One of the heuristic methods for this problem is the greedy algorithmwhich selects a vertex of minimum degree, deletes that vertex and all of its neighbours from the graph, and repeats this process until the graph becomes empty. Apart from that further a heuristic involved in the proposed algorithm is, if more than one vertices are with same (minimum) degree then select a vertex, which removal (along with its neighbours) causes removal of maximum number of edges toward the remaining graph. The performance ratio of an algorithm is defined to be the worst-case ratio of the
size of the optimal solution to the size of the algorithm's solution. So far many considerable performance ratios have been determined for the independent set problem, e.g. (d+1)/2 and (2d+3)/5 in [3], where d is average degree of vertices in the input graph. The independent set problem remains NP-hard on graphs of bounded maximum degree, but approximation becomes considerably easier. In fact, any algorithm that finds a maximal independent set has a performance ratio of $(\Delta+1)/2$, if $\delta\geq 2$. [2] presents an algorithm schema that involves removing small cliques from the graph. For graphs with few disjoint cliques, it can manually remove all the cliques and find the promised improved solution on the remainder.

Related Works:

A performance ratio of $(\Delta \log\log\Delta/\log\Delta)$ is given in [6] using subgraph removal technique. [3] achieves a performance ratio of $(\Delta + 2)/3$ for approximatingindependent sets in graphs with degree bounded by Δ, and also analyzes the algorithm when run in combination with a knownpreprocessing technique, and obtain an improved performance ratio of (2d+ 3)/5 on graphs with average degreed, improving on the previous best (d + 1)/2 of Hochbaum.Anefficient approximations of the weighted independent set problem in [4] obtained with performance ratios of $O(n(\log\log n/\log n)^2)$ and $(\Delta + 2)/3$, by partitioning cliques and hereditary induced sub graph.

Definition of Relevant Terminology

A simple graph G with n vertices consists of a set of verticesV, with |V| = n, and a set of edgesE, such that each edge is an unordered pair of distinct vertices. Neighbour of a vertex is represented by N(v). Δ, δ, and d is maximum, minimum, and average degree of a graph respectively. The adjacency matrix of G is an $n\times n$ matrix with the entry in row u and column v equal to 1 if $uv\in E$ and equal to 0 otherwise. An independent set I of G is a set of vertices such thatno two vertices in I are adjacent. A polynomial-time algorithm is one whose number of computational steps is bounded by a polynomial function of the size of the input. The class of such problems algorithms is denoted by **P**. For some problems, there are no known polynomial-time algorithms but these problems can be solved by nondeterministic polynomial-time algorithms. The class of all such problems is denoted by **NP**. The problem of finding a maximum independent set is known to be **NP**-Hard. The present approximation algorithm is, so far isknown, a promising candidate for the task.

Proposed Approximation Algorithm:

Consider a graph G (V, E), where d(v) is degree of vertex $v \in V(G)$,N (v) is neighbours of the vertexv. Purpose is to find the maximal (as much close as possible to maximum) independent set, and algorithm is given below.

Modified Greedy (G)

I←ø // I is a set of independent set, initialize with zero.

while $V(G) \neq \emptyset$ do

choose v such that $d(v)=MIN_{\omega\in V(G)}d(\omega)$

if more than one vertex having same Minimum

degree if min degree is zero or one, then just select it else choose v such that sum of edges of its adjacent vertices toward the remaining graph is maximum.

4. I ← IU{v}
5. G← G – {v} U N (v)

end

4. Output I

Time complexity:

An adjacency matrix $A_{n(n+1)}$ is maintained for the graph, where rows and columns range from o to n-1, and o to n respectively, last $(n+1)^{th}$ column is to count of 1's in each row, line 1 iterates |I| number of times. Line 2 takes at most O(n) time to find a list of minimum degree vertices (traverse the last column of the adjacency matrix). From line 3, let k1,k2,...kl (o≤ l< n) vertices are of same degree. Let a particular vertex in the list of obtained minimum degree vertex is having adjacent vertices a1,a2...ai (o≤i ≤ Δ), then find MAX{∑d(ai) - ∑A[ai][aj], where i≤ j≤ Δ }. Let the obtained (from MAX function) vertex is kj, set A[kj][n+1] and for all its adjacent A[as][n+1] to -1 . And also for o≤ i< n A[i][as]=o and A[i][n+1] - =1, this is how it takes time $O(n.\Delta^2)$. This process (in else part) all together would occur at most |I| times (according to amortized analysis). Line 4, union of one element to a set of elements takes constant time. Line 5 is accomplished by line 3 itself. This is how total time complexity required is $O((|I|n\ \Delta^2)$. This is at most O (n^4).

After applying the algorithm to most of the known graphs, The Tetrahedron, The Kuratowski Bipartite Graph $K_{3,3,}$ The Octahedron, The Bondy-Murty Graph $G_{1,}$ The Wheel Graph $W_{8,}$ The Cube, The Petersen Graph, The Bondy-Murty graph $G_{2,}$ The Grötzsch Graph, The Herschel Graph, The Icosahedron, The Bondy-Murty graph $G_{3,}$ The Bondy-Murty graph $G_{4,}$ The Ramsey Graph $R(4,4)$, The Folkman Graph, The Dodecahedron, The Tutte-Coxeter Graph, The Thomassen Graph, The Berge Graph, The Witzel Graph,Frucht graph, the proposed algorithm gives exact solution as it is obtained in [1] in order of $O(n^8)$ time.

Performance Bound:

As per the approximation algorithm proposed, first all zero degree and pendant vertices (with removal of its adjacent) are chosen, having the remaining graph with δ≥2, upper bound of independent set can be derived. First time a vertex with minimum degree δ is chosen, then onwards, if remaining number of vertices are even then there must be a vertex with degree one at least, so

n≥ (1+ δ) + (|I|-1)(1+1).....(1)

or, if remaining number of vertices are odd, then

n≥ (1+ δ) + (|I|-2)(1+1)+1 .....(2)

where, |I| is size of independent set.

From Eq (1) and (2)

|I|≤ (n+2-δ)/2 ..... (3)

Lower bound is determined in [2] as |I|≥n/Δ+1. An improved lower bound is derived here, as follows. As the greedy algorithm suggests, first time, the vertex with minimum degree (δ) has to be selected, then after there must be some vertex with degree Δ-1 (in worst case) available. So the |V(G)| (=n) can be written as

n ≤ (1+ δ)+(1+ Δ-1)+ (1+ Δ-1)+...(|I|-1) times

|I|≥((n-1+ Δ- δ)/ Δ) or considering |I| as integer,

|I|≥((n+Δ- δ)/ Δ) .....(4)

For δ≥2 (after choosing and removing all zero degree and pendent vertices), the derived lower bound appears very close to the exact value of independent set.

The performance ratio of such an algorithm is defined to be the worst-case ratio of the

size of the optimal solution to the size of the algorithm's solution. An algorithm in [3] with a known preprocessing technique obtains an improved (2d+ 3)/5 performance ratio on graphs with average degree d, improving on the previous best (d + 1)/2 of Hochbaum. Here, the performance ratio (R) is derived as

$$R = \frac{(n - \delta + 2)/2}{(n + \Delta - \delta)/\Delta}, \text{..... where } \Delta \geq \delta \geq 2$$

$$= \frac{\Delta}{2}\left(\frac{1 - \frac{\delta - 2}{n}}{1 + \frac{\Delta - \delta}{n}}\right) \text{..... ....(5)}$$

Theperformance ratios, obtained from various papers are compared and tabulated in the Table1(in Appendix I). The last column is obtained by this paper. It is found that when value of Δ and δ are close and comparable with n then it gives a almost constant performance irrespective of value of n.

Conclusion

The proposed algorithm provides an optimal solution (in order to find Maximum Independent set) for almost all known graphs with time complexity as at most O (n^4). This algorithm is verified with many of known graphs like, The Tetrahedron, The Kuratowski Bipartite Graph $K_{3,3,}$ TheOctahedron, The Bondy-Murty Graph $G_{1,}$The Wheel Graph $W_{8,}$ The Cube, The Petersen Graph, The Bondy-Murty graph $G_{2,}$ The Grötzsch Graph, The Herschel Graph, The Icosahedron, The Bondy-Murty graph $G_{3,}$ The Bondy-Murty graph $G_{4,}$ The Ramsey Graph $R(4,4)$, The Folkman Graph, The Dodecahedron, Frucht graph, The Tutte-Coxeter Graph, The Thomassen Graph etc.

References

1. Ashay Dharwadker,The Independent Set Algorithm, published by Amazon in 2011
2. Magnus M. Halldorsson, Jaikumar Radhakrishnan, Improved Approximations of Independent Sets in Bounded-Degree Graphs via Subgraph Removal, Nordic Journal of Computing 1(1994), 475-492.
3. M.M. Halld6rsson, and J. Radhakrishnan, Greed is Good: l Approximating Independent Sets in Sparse and Bounded-Degree Graphs, 1997 Spdnger-Verlag New York Inc.
4. Magn´us M. Halld´orsson, Approximations via Partitioning, 14 March 1995.
5. K. Jansen,J. Rolim,editors, Approximations of independent sets in graphs. 1998.
6. Magn´us M.Halld´orsson, Approximations of Weighted Independent Set and Hereditary Subset Problems, Science InstituteUniversity of Iceland, vol. 4, no. 1, pp. 1–16, 2000.

Appendix I

Table1: Comparison among Various Performance Ratios.

n (=|V|)	**Δ(=n/2)**	**δ(=n/3)**	**d (=(Δ+δ)/2)**	**(d+2)/3**	**(2d+3)/5**	**Δ log log Δ/log Δ**	**(Δ+3)/4**	**(Δ/2)*(1-(δ-2)/n)/(1+(Δ-δ)/n)**
10	10	6.67	8.335	3.45	3.268	0	3.25	3.29
66	33	22	27.5	9.83	9.4	3.94	9	9.86
200	100	66.67	83.335	28.45	27.268	15.05	25.75	29
512	256	170.67	213.335	71.78	68.868	40.58	64.75	73.57
1024	512	341.33	426.665	142.89	137.132	81.8	128.7	146.71
204	1024	682.67	853.335	285.11	273.668	162.81	256.7	293

N	Δ (=n-1)	δ(=n-10)	d (=(Δ+δ)/2)	(d+2)/3	(2d+3)/5	Δ log log Δ/log Δ	(Δ+3)/4	(Δ/2)*(1-(δ-2)/n)/(1+(Δ-δ)/n)
20	19	10	14.5	5.5	4.6	1.59	5.5	3.93
32	31	22	26.5	9.5	9.4	3.61	8.5	4.54
100	99	90	94.5	32.17	36.6	14.89	25.5	5.45
512	511	502	506.5	169.5	201.4	81.64	128.5	5.88
1024	1023	1014	1018.5	340.17	406.2	162.65	256.5	5.94
204	2047	2038	2042.5	681.5	815.8	321.46	512.5	5.97

N	Δ (=n-10)	δ(=n-50)	d (=(Δ+δ)/2)	(d+2)/3	(2d+3)/5	Δ log log Δ/log Δ	(Δ+3)/4	(Δ/2)*(1-(δ-2)/n)/(1+(Δ-δ)/n)
500	490	450	470	157.33	180.6	78.28	123.2	23.59
100	990	950	970	324	380.6	157.47	248.2	24.75
1500	1490	1450	1470	490.67	580.6	235.48	373.2	25.16
200	1990	1950	1970	657.33	780.6	312.7	498.2	25.36
300	2990	2950	2970	990.67	1180.6	465.44	748.2	25.57
500	4990	4950	4970	1657.33	1980.6	766.4	1248.	25.74

Krishna Singh, krishnasingh@rgukt.in, L. Govinda, govinda.891@gmail.com,
Dept. of CSE, RGUKT-IIIT, Nuzvid (AP), India