BHOSLIB: Benchmarks with Hidden Optimum Solutions for Graph Problems (Maximum Clique, Maximum Independent Set, Minimum Vertex Cover and Vertex Coloring)

------ Hiding Exact Solutions in Random Graphs

If you have any comments or need more instances for the following benchmarks, please send me an email.

Maximum Independent Set (MIS) and Minimum Vertex Cover (MVC)

Given a graph G, an independent set is a subset S of vertices in G such that no two vertices in S are adjacent (connected by an edge). The maximum independent set problem is to find an independent set with the largest number of vertices in a given graph. This problem is NP-hard, and as such, it is considered unlikely that there exists an efficient algorithm for solving it. A vertex cover (or hitting set) is a subset S of vertices such that each edge of G has at least one of its endpoints in S. It is easy to see that given an independent set of a graph, all vertices not in the set form a vertex cover. The minimum vertex cover problem is to find a vertex cover with the smallest number of vertices. Finding challenging benchmarks for the maximum independent set problem (or equivalently, the minimum vertex cover problem) is not only of significance for experimentally evaluating the algorithms of solving this problem but also of interest to the theoretical computer science community. The MIS and MVC benchmarks presented here are transformed from forced satisfiable SAT benchmarks of Model RB, with the set of vertices and the set of edges respectively corresponding to the set of variables and the set of binary clauses in SAT instances (Thanks to Klaus D. Witzel for bringing our attention to this transformation). In fact, based on Model RB and this transformation, we can propose a simple random graph model as follows:

1. Generate n disjoint cliques, each of which has n^{α} vertices (where $\alpha > 0$ is a constant);

2. Randomly select two different cliques and then generate without repetitions $pn^{2\alpha}$ random edges between these two cliques (where $0 < p < 1$ is a constant);

3. Run Step 2 (with repetitions) for another $rn\ln n - 1$ times (where $r > 0$ is a constant).

It is easy to see that for the graph model above, the size of maximum independent sets is at most n. Interestingly, determining if such a upper bound can be reached is equivalent to determining the satisfiability of the corresponding CSP and SAT instances of Model RB, and there is a one-to-one mapping between the solutions of these two problems. To hide an independent set of size n in the instances of this graph model, we first select a vertex at random from each disjoint clique to form an independent set of size n, and then in the above process of generating random edges (Step 2), no edge is allowed to violate this maximum independent set. From the results of the paper "Many Hard Examples in Exact Phase Transitions with Application to Generating Hard Satisfiable Instances", it is expected that Model RB can also be used to generate hard instances for graph algorithms (by use of the above graph model and the hardness of phase transitions, i.e. choosing appropriate values of α, p and r). It is also hoped that the method of hiding solutions based on Model RB can have potential applications in cryptography. Recently, it has been shown (both experimentally and theoretically) that unlike random 3-SAT, it is quite natural and easy to hide solutions in random CSP and SAT instances of Model RB. For more details, please see an IJCAI-05 paper entitled "A Simple Model to Generate Hard Satisfiable Instances".

Note: Unless specified explicitly, all the benchmarks provided on this page are expressed in the ASCII DIMACS graph format.

- frb30-15-mis.tar.gz (205 KB): 450 vertices (30 cliques) - 5 instances, with the size of the MIS = 30 and the size of the MVC = 420

- [frb35-17-mis.tar.gz](#) (319 KB):   595 vertices (35 cliques) - 5 instances, with the size of the MIS = 35 and the size of the MVC =   560
- [frb40-19-mis.tar.gz](#) (470 KB):   760 vertices (40 cliques) - 5 instances, with the size of the MIS = 40 and the size of the MVC =   720
- [frb45-21-mis.tar.gz](#) (666 KB):   945 vertices (45 cliques) - 5 instances, with the size of the MIS = 45 and the size of the MVC =   900
- [frb50-23-mis.tar.gz](#) (930 KB): 1150 vertices (50 cliques) - 5 instances, with the size of the MIS = 50 and the size of the MVC = 1100
- [frb53-24-mis.tar.gz](#) (1094 KB): 1272 vertices (53 cliques) - 5 instances, with the size of the MIS = 53 and the size of the MVC = 1219
- [frb56-25-mis.tar.gz](#) (1279 KB): 1400 vertices (56 cliques) - 5 instances, with the size of the MIS = 56 and the size of the MVC = 1344
- [frb59-26-mis.tar.gz](#) (1478 KB): 1534 vertices (59 cliques) - 5 instances, with the size of the MIS = 59 and the size of the MVC = 1475

A brief description of the ASCII DIMACS graph format:

- There are 0 or more lines starting with c at the top of the file which are comment lines and can be ignored.

- Following the comment lines, there is a line with the form "p edge V E" which specifies the size of the graph where V and E are the number of vertices and edges respectively.

- The remaining of the file is a list of lines starting with e which indicate the edges in the graph (e.g. the line "e 1 3" indicates that there is an edge between vertex 1 and vertex 3).

Maximum Clique and Vertex Coloring

Given a graph G, a clique is a subset S of vertices in G such that each pair of vertices in S is connected by an edge. The maximum clique problem refers to the problem of finding a clique with the largest number of vertices in a given graph. Since a clique is an independent set in the complementary graph, the maximum independent set problem and the maximum clique problem (which is one of the first shown to be NP-hard and has been extensively studied in graph theory and combinatorial optimization) are essentially equivalent. The following benchmarks are the complements of the graph instances above.

- [frb30-15-clq.tar.gz](#) (914 KB):   450 vertices - 5 instances, with both the maximum clique size and the chromatic number = 30
- [frb35-17-clq.tar.gz](#) (1599 KB):   595 vertices - 5 instances, with both the maximum clique size and the chromatic number = 35
- [frb40-19-clq.tar.gz](#) (2647 KB):   760 vertices - 5 instances, with both the maximum clique size and the chromatic number = 40
- [frb45-21-clq.tar.gz](#) (4085 KB):   945 vertices - 5 instances, with both the maximum clique size and the chromatic number = 45
- [frb50-23-clq.tar.gz](#) (6444 KB): 1150 vertices - 5 instances, with both the maximum clique size and the chromatic number = 50
- [frb53-24-clq.tar.gz](#) (7979 KB): 1272 vertices - 5 instances, with both the maximum clique size and the chromatic number = 53
- [frb56-25-clq.tar.gz](#) (9774 KB): 1400 vertices - 5 instances, with both the maximum clique size and the chromatic number = 56
- [frb59-26-clq.tar.gz](#) (11890 KB): 1534 vertices - 5 instances, with both the maximum clique size and the chromatic number = 59

Note: For vertex coloring of the above instances, what is hard to solve is not to find the optimum colorings, but to prove the optimality of such solutions. For example, it should be very hard to prove that 59 is fewest number of colors to color the instance of frb59-26-1.clq.

Below are two complementary graph benchmarks generated using the same method and the same values of parameters (α, p and r) as above. The clique benchmark has a hidden maximum clique of size = 100 and, accordingly, the MIS benchmark has a hidden maximum independent set of size = 100. Based on theoretical analysis and experimental results of smaller instances, I conjecture that in the next 20 years or more (from 2005), these two benchmarks can not be solved on a PC (or alike) in a reasonable time (e.g. 1 day). Even so, it would be very interesting to see how large the gap is between the optimum value and what are obtained by various solvers. It is also of interest to see how this gap is going to be bridged with the development of algorithms and computers. Starting from these ideas, I hope that anyone of interest can have a try on solving the two benchmarks and, if possible, please inform me of the result you obtain (including the size of the largest clique or the largest independent set you find, how much time is used, on what kind of computer and by what solver). If agreed, your result will be posted as a news here and a link could also be added to your website (if there is one).

- **frb100-40.clq.gz** (17384 KB): **4000 vertices and 7425226 edges, with the maximum clique size = 100**
- **frb100-40.mis.gz** (1349 KB): **4000 vertices and 572774 edges, with the size of the MIS = 100 and the size of the MVC = 3900**

The DIMACS binary format, an alternative way for storing a graph, is space efficient and is used in many solvers. The above two benchmarks in this format are respectively available as frb100-40.clq.b (979KB) and frb100-40.mis.b (979KB).

News

- Dec.1, 2014: Using a stochastic local search solver (called ULSA), Christopher D. Rosin at Amara Health Analytics found **independent sets of size 99** on the frb100-40.mis instance, in under 24 hours of runtime on a Linux server with a 3.4GHz i7-4770 CPU. ULSA's single-threaded average time to find 99 on this CPU is 29.3 hours.

- Mar. 20, 2010: Using a local search solver (called EWLS) on a 3GHz/4GB computer, Shaowei Cai and Kaile Su at Peking University found **a vertex cover of 3902 vertices** (or equivalently, an independent set of 98 vertices) for the above MIS benchmark of 4000 vertices. More specifically, the authors performed 100 independent trials on this instance. 4 runs find a 3902-sized vertex cover, or equivalently, a 98-sized independent set with the average time of 2823.05 seconds and the fastest one did so in **1239.87** seconds. Of the other 96 runs, 59 found a 3903-sized solution and 37 found a 3904-sized solution.

- Jul. 4, 2007: Using a stochastic local search solver (called COVER) on a 2.13GHz/2GB computer, Silvia Richter at Griffth University found **a vertex cover of 3903 vertices** (or equivalently, an independent set of 97 vertices) for the above MIS benchmark of 4000 vertices in **71.09** seconds. More specifically, the author ran the solver 100 times with different seeds. Of those 100 runs, 25 found a solution of size 3903, the other runs all found a solution of size 3904. Of the 25 successful runs that found the better solution, the quickest one did so in 71.09 seconds while the median run-time was 1193.92 seconds.

- Jan.19, 2006: In 2005, according to weblog statistics, this webpage was visited 5147 times (by people from 54 countries) and the benchmarks on this page were downloaded 1472 times (including 169 times for the above two benchmarks of 4000 vertices). Most visitors were from USA, China, Canada, India, Germany, UK, France, Italy, Australia and Israel (about half of them came through Google).

- Jul. 28, 2005: Using a local search SAT solver (called wsatcc) on a Pentium IV 3.4GHz/512MB computer, Lengning Liu at the University of Kentucky found **a vertex cover of 3904 vertices** (or equivalently, an independent set of 96 vertices) for the above MIS benchmark of 4000 vertices in **3743.16** seconds.

- Feb.22, 2005: The above two benchmarks of 4000 vertices were generated and made available online.

List of Papers That Use Graph Benchmarks Based on Model RB (Updated: Mar. 4, 2009)

Other Benchmarks Based on Model RB

Forced Satisfiable CSP and SAT Benchmarks of Model RB

Pseudo-Boolean (0-1 Integer Programming) Benchmarks with Hidden Optimum Solutions

Benchmarks with Hidden Optimum Solutions for Set Covering, Set Packing and Winner Determination

Weighted Max-2-SAT Benchmarks with Hidden Optimum Solutions (New!)

Back to Ke Xu's homepage.

Date Created: April 20, 2004. Last Updated: Dec. 1, 2014.