Associate Editor's Report

"A penalty-evaporation heuristic in a decomposition method for the maximum clique problem"

October 2002

The reviewers were impressed with the computational results but feel the overall investigation of the algorithm needs strengthening, particularly with regard to the decomposition method. I recommend that the paper be returned for major revisions and re-review. The authors should undertake the additional research into the effectiveness and behavior of the decomposition aspect of their heuristic, with appropriate additional computational comparisons, as well as make the other clarifications and changes suggested. I am sorry for the appearance of some of the reviews – email apparently did not fully preserve the formatting.

The article describes a method of decomposing a graph G into several sub-graphs, in which the maximum clique problems (MCP) are solved and show that they can gain the maximum clique (MC) of the whole graph G by the solutions of the sub-graphs. This result holds whenever exact algorithms for the MCP are used inside the sub-problems. They also provide a new heuristic, so-called penalty-evaporation (P-E) heuristic, for finding large maximal cliques. The combination of the decomposition with the P-E heuristics shows promising results in the experimental section.

I can recommend the paper for publication after several (sometimes major) adjustments are made:

I have the following suggestions:

on p.4-p2-l2 the authors write about exponentially large number of cliques. first this is not true in general, second they should write exponentially in _what_ variable and third they should refer to some literature (e.g. moon/moser(65), On cliques in graphs, Isr. J. Math., 3, 23--28)
it should be mentioned that solving the maximum independent set problem is equal to solving the MCP in the complementary graph (the terms are mixed up in the lit. rev.)

on p.6,p3: they describe genetic approaches to the MCP in this context the work of I.M. Bomze and V. Stix(99), Genetic engineering via negative fitness: Evolutionary dynamics for global optimization, Annals of Oper. Res., 89, 297--318 would fit in, because it combines the genetic approach mentioned in this paragraph with a strategy (mentioned earlier on p.4) in order to escape local inefficient solutions, which delivers an exact method for the MCP.

on p.6,sec 3: The decomposition section comes out of the blue without motivation after the literature review. maybe on or two sentences could help (but see also below regarding the structure of the paper)

on p.6,p2,l 1-3: "at each iteration...". It is not clear that the large clique is especially constructed, such _that_ its vertices do not belong to any other larger one.
algorithms: the algorithms are not very clear. at least there should be a line saying what is the result of the algorithm and in which variable can it be found.

on p.6,theorem: "(i.e. the algo..." should be like "(i.e. the

algorithm finds one maximum clique inside Gâ€™(i)"
cont.: "then the decomposition ... generates an optimal solution" What is the optimal solution (again maximum clique should be used) and where can this solution be found? What is the output of the algo? (see above)

on p.6,proof: at first sight it seemed to me that the proof consists of two lines only. the paragraphs should be held together or a end-of-proof symbol would be nice.
the algorithm itself reminds me of the algorithm of babel (reference 2). babel also iterates through vertices and looks inside the neighborhood of them for maximum cliques. There are some other differences but maybe the author should point them out.
secondly what i am really missing is a recursive approach of the decomposition algorithm (as babelâ€™s decomposition offers). step 2.1 of the algorithm (find the MC inside Gâ€™(i)) is practically impossible when using exact algorithms, because the decomposition method does only a single decomposition on the first level. In dense graphs the decomposed graphs Gâ€™(i) will have almost the same order than the original one. therefore i ask the authors if they have thought about a recursive application of the decomposition in order to apply some exact methods once the graph is really smaller.
maybe thatâ€™s the reason why only heuristics are applied to the decomposition in the experimental section.

on p.7, algorithm: maybe only step 3 should be written down, because its the only step that differs from the prev. algo.
the P-E approach (section 4) is well done and there are a lot of interesting ideas in it. again on p11 the algorithm could be a bit more self- explanatory (what is the in/output)
all tables in the article should be more self-explanatory. e.g. i would suggest to write the numbers in table 5.1 5.2 ... in a percentage of success or failure instead of absolute numbers. the caption of the tables and the labels (especially tab 5.5 5.6 5.7) should be improved. it is very hard to find the information within the mixture of text and tables.

final comments:
because the main results and experiments concentrate on the P-E algorithm i would suggest to start with that one and incorporate the decomposition into the P-E algorithm afterwards. I do not think that the presented decomposition works well on exact methods because of the reasons mentioned above. I wander how the P-E algo. would succeed using other easy decompositions like e.g. the one suggested by babel (on first level only).

it should be distinguished between finding the MC and proofing that the found solution is the MC, which is much harder. all algorithms described in this article are only able to find the MC. If the size of the MC, however, is unknown the deliver only a lower bound on it.

Referee's report

Title

A Penalty-Evaporation Heuristic in a Decomposition Method for the Maximum Clique Problem.

Authors

General comments

The paper is very well written and gives a short and precise introduction to the topic. Their contribution is very well explained and they get impressive results. Their decomposition method clearly shows its benefits to improve the underlying heuristic. Their local search heuristic P-E alone performs almost as good as tabu search. Since tabu search often gets better results than P-E alone, it would have been interesting to know the performance of a decomp-tabu algorithm. Unfortunately this has not been discussed in the article.

The two proposed procedures are general enough to be adapted to other problems. Due to this fact and the results obtained the article is of great interest to the research field.

I recommend this article to be published without modification.

=20 =20 =20 =20 =20 =20 =20 =20 =20 =3D=3D=3D=3D=3D=3D=3D=3D=3D
=20

figures=3D=3D=3D=3D=3D=3D=3D=3D=3D=3D=3D=3D=3D=3D=3D=3D=3D=3D=3D=3D=3D=3D=
=3D=3D

Referee's report on the paper:

A penalty-evaporation heuristic in a decomposition method for = the maximum clique problem

The authors present a new heuristic approach to the solution of the = maximum clique problem along with a decomposition scheme to reduce the = size of the processed problems. The basic algorithm can be viewed as a = modified tabu search; however, rather than prohibiting the selection of = a specific move, such selection is made more difficult by adding a = penalty term to its fitness value. Large penalties are equivalent to a = full prohibition. The penalty associated to a move is reduced at each = iteration (*evaporation*). This "penalty-evaporation" heuristic is = applied to the solution of the subproblems obtained by applying the = decomposition scheme to the original instance. The decomposition is = based on the following, simple principle: if C is (the vertex set of) = a clique, and no larger cliques are contained in the subgraph induced by = C and its neighbourhood, then C can be removed, and the search can = be carried on in the residual graph. This scheme is exact, but can be = relaxed (with minor amendments) by contenting ourselves with finding = large cliques. The algorithm has been tested over the DIMACS instances, = and the computational results appear to be competitive with the best = algorithms in the literature. =20

In my opinion the paper can be accepted after revision. Even though the = ideas developed are not too deep, still the algorithm favourably compare = with the best algorithms presented in the literature; also, the = experiments have been carefully carried out on the DIMACS test-bed, = which is good. The bibliography section is sufficiently rich. Both the = English usage and the mathematical formalism should be improved in the = revised version.=20 A main observation concerns the decomposition mechanism, which plays a = crucial role in enhancing the algorithm. However, it is unclear to me = how this decomposition has an effect on the difficult instances (the = dense ones). Specifically, while executing Step 2 for the first time the = algorithm will probably explore a large portion of the graph. In other = words, if the graph is dense, it is very likely that a largest clique = contains one of the vertices in the first clique(s). I think some = computational results on a pair of dense instances may be of help. In = particular, the authors should report the size of the cliques found at = each iteration and the size of the neighborhood of such cliques. Also, a = set of experiments (even on randomly generated graphs) to relate the = efficacy of the decomposition versus the density of the instances should = be included. =20

It follows a list of specific comments.

1. Page 1, Line -14: The decomposition scheme is not a solution = approach. More correctly the authors introduce one=20 solution approach along with an effective decomposition scheme.=20

2. Page 1, Line -6: I do not think that the definition = *greedy-like* fits with the presented algorithm. In fact, both the formal = definition (based on matroid theory) as well as the practical and = standard usage of the term involve some type of irrevocable choice (i.e. = an element is included in the solution and never removed). On the other = hand, most of known heuristics include some type of greedy choice (i.e. = an element maximizing some weighing function is temporarily included in = the solution). See for example [?].

3. Page 2, Line 6: this looks more like an example of application of = graph coloring; many other (more fitting) applications are at hand. =20

4. Page 2, Line 16: "... There exists ..." change to "... There = exist ..." =20

5. Page 3, Line 10: "... ajusting ..." change to "... adjusting ..." = =20

6. Page 4, Line -17: the expression "different parts of the graph" = (also used other times in the sequel), in this context, is too sloppy = and misleading. Still being informal, but a bit more precise, one could = use the expression "solution space" instead of "graph". =20

7. Page 5, Line 1-4. Actually, this is not a way to *avoid* the = use of a tabu list: it is only a smart way to *represent* a tabu = list - as observed by the authors in [21] (in the bibliography section = of the paper). There is a confusion here between an abstract data = structure and its implementation.=20

8. Page 6, Line 15: the definition of neighborhood (typically denoted = by $N(i)$) is non standard. In general, vertex i does not belong to = its neighborhood. Also, the neighborhood is not a graph, but a set of = vertices. =20

9. Page 6, Line 17: "... are adjacent ..." change to "... is adjacent = ...".

Reviewer 3
3 of 3

10. Page 6, Line -6: the proof of the main theorem is a bit involved. = I propose something like " Let $\bar{k}$ be the smallest k such that = $C^* \cap C^k \neq \emptyset$: then $|C^{\bar{k}}| = 3D|C^*|$. In fact, = (denoting by G^k the residual graph at iteration k), C^* is = completely contained in $G^{\bar{k}}$." =20

11. Page 8, Line 18: see note Page 1, Line -7.=20

12. Page 9, Line 8: "... proportionnal ..." change to "... = proportional ..." =20

13. Page 12, Line 2: what is an "interesting part of the graph"? Be = more precise (see also note Page 4, Line -17).=20

14. Page 12, Line 21: here and later, "best" and "optimal" stands also = for "best known". Actually, this is later specified but I think the = authors should mention it earlier. =20

15. Page 15, Line -16: "... independant ..." change to "... = independent ..." =20

16. Page 20, Line 6: "... small values ..." change to "... small = absolute values ..." =20

References

[1] Local Search In Combinatorial Optimization. *E. Aarts and J.K. Lenstra editors*, John Wiley & Sons, = 1997.=20