Published in J. ALGORITHMS 2, 393-405 (1981)

The NP-Completeness Column: An Ongoing Guide

DAVID S. JOHNSON

Bell Laboratories, Murray Hill, New Jersey 07974

1. STATEMENT OF PURPOSE

This is the first edition of a column which is to appear quarterly in the *Journal of Algorithms*. When Mike Garey and I published our book *Computers and Intractability: A Guide to the Theory of NP-Completeness,* W. H. Freeman & Co., San Francisco, 1979 (hereinafter referred to as ''[G&J]''), we knew that our list of some 300 NP-complete and NP-hard problems could not remain ''state-of-the-art'' for long. We were not prepared, however, for the flood of new results that has continued to pour forth in the last two years. Half of the questions in our open problem section have been resolved, and hundreds of interesting new NP-complete problems have been discovered, with new ones arriving almost daily.

There seems to be no easy way for the average researcher or practitioner to keep up, and even certain veteran list-makers are having trouble keeping track of developments. The range of journals in which new results appear has considerably broadened, and the results are often buried in an appendix or only mentioned in passing. Also, more and more often, results are being proved but *not* published. Journals are growing less willing to publish papers consisting of NP-completeness results, unless the problems considered are very significant and/or the paper also contains other work, such as the analysis of algorithms for special cases or for related problems. Researchers themselves are often reluctant to publish individual NP-completeness results, especially when the proofs use more-or-less standard techniques. Consequently, many interesting results are now only transmitted by word-of-mouth, and hence are known only within small circles. When I travel, I find that practically everyone I talk to has a new result to relay or wants to check on some rumor he has heard.

Thus, with encouragement from many of my colleagues in the field, I have decided to initiate a quarterly column that can serve as a clearinghouse for new NP-completeness results (and new polynomial-time solvable subcases of NP-complete problems). The column will provide a forum for researchers wishing to publicize their results prior to (or instead of) more formal publication, and should be of use to anyone wishing to keep up with the ''state-of-the-art'' of NP-completeness.

There is, of course, an ulterior motive in my willingness to undertake this column. Mike Garey and I are already beginning to receive inquiries as to when a second edition of our book (with an updated list) might appear. By writing this column I can spread out the work of updating, and also postpone the *need* for a second edition until a time when the field is no longer growing so explosively.

2. The Nature of the Column

Although the column may evolve over time, initially it will contain the following types of material:

1. Descriptions of new NP-complete (or NP-hard) problems, together with comments about related results and the complexity of special cases, in the format of [G&J].

2. Brief summaries of papers in specialized areas, where only a few ''typical'' results may qualify for explicit mention in a ''general interest'' column, but much more is known. A noteworthy example of this type of paper is the survey of scheduling results prepared by Lageweg *et al.* [43].

3. Open problems submitted by readers.

4. Updates to the references in [G&J] and this column, reporting on the formal publication of papers originally cited as ''personal communication,'' ''unpublished manuscript,'' or ''to appear.''

5. Occasional reports on new theoretical developments concerning NP-completeness and related issues.

In general, proofs will not be included, although exceptions may be made for particularly elegant or instructive ones that might not otherwise be published. (Either I or my ''trusted colleagues'' will verify the proofs of all NP-hardness proofs cited, and files will be maintained containing these proofs, or sketches of them). At least initially, while I am catching up on the backlog, I shall organize the columns around specific themes. For instance, an early column might survey the recent rash of complexity results concerning VLSI design.

The reader of this column will be assumed to be familiar with the basic theory of NP-completeness, as described in [G&J] or other works, such as [3,33,37,38]. Specific familiarity with [G&J] will not be assumed, although cross-referencing to that book will be included when appropriate.

3. Submissions to the Column

Anyone having results or open problems he wishes mentioned in the column should send them to David S. Johnson, Room 2C-355, Bell Laboratories, Murray Hill, NJ 07974. For new results the types of submissions are, in order of preference:

1. References to published papers containing results not already cited in [G&J]. Please include a brief description of the nature of the results, and the precise place in the paper where the results are presented. Include a reprint if possible.

2. Technical reports or unpublished manuscripts containing results as above. Again, please include a brief description and a precise location.

3. Informal notes on new results, stating the claimed results precisely and giving at least some indication of the proof.

4. Rumors, together with some idea of how they might be tracked down.

In the case of unpublished results, please state explicitly that you would like the results mentioned in the column.

Proposed open problems should be described precisely, and should be accompanied by an explanation of the motivation behind them, including, where possible, reference to relevant published works. In addition, suggestions, comments, and corrections (both to the column and [G&J]) are welcome.

4. Open and Closed Problems in NP-Completeness

I shall devote the technical portion of this first column to a discussion of the progress that has been made on the twelve open problems presented at the end of [G&J]. As mentioned above, six of these have been resolved. The split is even: three have been shown to be solvable in polynomial time and three have been proved NP-complete. In addition, there have been new developments related to all the other six. In order to maintain some suspense, I shall discuss the problems in their original order.

[1] GRAPH ISOMORPHISM

INSTANCE: Two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$.

QUESTION: Are G_1 and G_2 isomorphic, i.e., is there a one-to-one onto function $f: V_1 \rightarrow V_2$ such that $\{u,v\} \in E_1$ if and only if $\{f(u), f(v)\} \in E_2$?

Comment. This problem remains open, although some significant new subcases have been shown to be solvable in polynomial time: graphs embeddable in the projective plane [51], graphs embeddable in a surface of genus K, for every fixed K [20,58], graphs with maximum vertex degree K, for every fixed K [55], and ''cone graphs'' of bounded degree (as defined in [31] - note that the ''degree'' of a cone graph is not its vertex degree), using the techniques from [55]. In addition, ''colored'' graphs, in which each vertex is assigned a color, can be tested for isomorphism in polynomial time if no color occurs on more than K vertices, for every fixed K [55]. A subexponential $O(n^{c \log n})$ algorithm has been found for tournaments [55]. Many of these latter results follow from polynomial-time algorithms for dealing with permutation groups, as described in [23]. They also were foreshadowed by various randomized algorithms for the same problems, such as in [5]. In the realm of probability, [7] presents a method for producing canonical labellings of graphs (and hence for testing isomorphism) which runs in expected linear time under the standard uniform distribution of graphs. The best worst case running time for the general problem remains exponential, but there have been significant improvements over the naive $O(n!)$ bound, where n is the number of vertices: M. K. Goldberg gives an $O(c^n)$ algorithm in [26], and V. N. Zemlyachenko improves on this to $O(c^{n^{2/3}})$ [6,71], using techniques from [55]. Finally, the set of problems which have been proved polynomially equivalent to general graph isomorphism has been considerably broadened [9,12,13,14], although the transformations involved may not always be able to preserve the exponents in the running times of the above algorithms.

[2] SUBGRAPH HOMEOMORPHISM (FOR A FIXED GRAPH H)

INSTANCE: Graph $G = (V,E)$.

QUESTION: Does G contain a subgraph homeomorphic to H, i.e., a subgraph $G' = (V',E')$ that can be converted to a graph isomorphic to H by repeatedly removing any vertex of degree 2 and adding the edge joining its two neighbors?

Comment. This problem remains open, but the ‘‘fixed vertex’’ version of the problem (the input specifies exactly which vertex of G is to correspond to each vertex of H) has been completely classified by Fortune, Hopcroft, and Wyllie [22] for *directed* graphs. It is polynomial-time solvable if H is a fixed graph all of whose arcs share a common tail, or all of whose arcs share a common head. For all other fixed graphs H the fixed-vertex directed SUBGRAPH HOMEOMORPHISM problem is NP-complete. If the input graph G is an acyclic directed graph, then the problem is solvable in polynomial time for each fixed H [22], although the problem remains NP-complete if H is allowed to be specified as part of the instance, as follows from [ND38] in [G&J]. The problem is also solvable in polynomial time for each fixed H if G is restricted to reducible flow graphs [30]. With respect to the undirected version of the problem, new polynomial time algorithms for the case where H consists of two independent edges have been presented in [59,64], with the latter giving a particularly concise characterization. The original two edge and triangle results mentioned in [G&J] have appeared formally in [65,44]. With respect to the ‘‘non-fixed vertex’’ versions of the problem, [52] presents a polynomial time algorithm for $H = K_4$ in the undirected case and [22] classifies a number of H (both polynomially solvable and NP-complete) in the directed case, but much remains open (a polynomial time fixed vertex algorithm for H implies a polynomial time non-fixed vertex algorithm for H, but not vice versa). The special case where G is a tree can be solved in polynomial time even when H is not fixed but is part of the input [66].

[3] GRAPH GENUS

INSTANCE: Graph $G = (V,E)$ and a non-negative integer K.

QUESTION: Can G be embedded on a surface of genus K such that no two edges cross one another?

Comment. Remains open for general K, but can be solved in polynomial time for every fixed K [21]. A detailed algorithm for the case of $K = 1$ and cubic graphs is given in [19]. This work on genus led to the isomorphism algorithms for graphs of fixed genus mentioned above.

[4] CHORDAL GRAPH COMPLETION

INSTANCE: Graph $G = (V,E)$ and a positive integer K.

QUESTION: Is there a superset E' containing E of unordered pairs of vertices from V that satisfies $|E' - E| \leq K$ and such that $G' = (V,E')$ is chordal, i.e., such that for every simple cycle of more than 3 vertices in G', there is some edge in E' that is not involved in the cycle but that joins two vertices in the cycle?

Comment. This problem has been proved NP-complete by M. Yannakakis [68] via a transformation from OPTIMAL LINEAR ARRANGEMENT [GT42].

[5] CHROMATIC INDEX

INSTANCE: Graph $G = (V,E)$ and a positive integer K.

QUESTION: Does G have chromatic index K or less, i.e., can E be partitioned into disjoint sets $E_1, E_2, \ldots, E_k$, with $k \leq K$, such that, for $1 \leq i \leq k$, no two edges in E_i share a common endpoint in G?

Comment. This problem has been proved NP-complete by I. Holyer [32], via a transformation from 3SAT, even for $K = 3$ and cubic graphs. Leven and Galil [50] have shown that NP-completeness also holds for every fixed $K > 3$, even if the graphs are required to be K-regular (a surprisingly non-trivial extension).

[6] SPANNING TREE PARITY PROBLEM

INSTANCE: Graph $G = (V,E)$ and a partition of E into disjoint 2-element sets $E_1, E_2, \ldots, E_m$.

QUESTION: Is there a spanning tree $T = (V,E')$ for G such that for each E_i, $1 \leq i \leq m$, either $E_i \subseteq E'$ or $E_i \cap E' = \emptyset$?

Comment. This problem can be solved in polynomial time by an algorithm due to L. Lovász [53,54], which can be used to solve the parity problem for arbitrary ‘‘representable’’ matroids, of which this is a special case (a matroid is representable if its independent sets correspond to the independent sets in some linear space). There can be no polynomial time algorithm for the general matroid parity problem if one is not allowed any knowledge about the structure of the matroid beyond that provided by a ‘‘black box’’ algorithm, i.e., an oracle that can tell (in one step) whether any given set is an independent set of the matroid [53]. (Recall that the 2-matroid intersection problem *can* be solved in polynomial time under such restrictions - see [45]).

[7] PARTIAL ORDER DIMENSION

INSTANCE: Directed acyclic graph $G = (V,A)$ that is transitive, i.e., whenever $(u,v) \in A$ and $(v,w) \in A$, then $(u,w) \in A$, and a positive integer $K \leq |V|^2$.

QUESTION: Does there exist a collection of $k \leq K$ linear orderings of V such that $(u,v) \in A$ if and only if u is less than v in each of the orderings?

Comment. This has been proved NP-complete independently by Lawler and Vornberger [47] and by M. Yannakakis [69], although the Yannakakis result is stronger. It shows that NP-completeness holds for every fixed $K \geq 3$, whereas the other result was only for arbitrary K. The proof is via a transformation from GRAPH 3-COLORABILITY [GT4]. The problem remains NP-complete even if the partial order contains no chain of length greater than one, so long as K is fixed at a value of 4 or more [69]. The problem for general partial orders was already known to be solvable in polynomial time when $K = 2$ [17,62].

[8] PRECEDENCE CONSTRAINED 3-PROCESSOR SCHEDULING

INSTANCE: Set T of unit length tasks, partial order $\lessdot$ on T, and a deadline $D \in Z^+$.

QUESTION: Can T be scheduled on 3 processors so as to satisfy the precedence constraints and meet the overall deadline D, i.e., is there a schedule $\sigma: T \rightarrow \{0,1,\ldots,D-1\}$ such that $t \lessdot t'$ implies $\sigma(t) < \sigma(t')$ and such that for each integer i, $0 \leq i \leq D-1$, there are at most 3 tasks $t \in T$ for which $\sigma(t) = i$?

Comment. This problem can be solved in linear time if $\lessdot$ is the disjoint union of an in-forest with an out-forest [16,25], and analogous polynomial time algorithms exist for every fixed number K of processors [25]. However, if K is arbitrary the problem becomes NP-complete even for such restricted precedence constraints [25]. (Recall that the problem can be solved in polynomial time if $\lessdot$ is an in-forest *or* an out-forest [34]). The problem can also be solved in polynomial time for every fixed K if $\lessdot$ is a *level order* (any two tasks with a common immediate predecessor or successor have identical sets of predecessors and successors) [67], and for arbitrary K if $<$ is an *interval order* (each task T_i corresponds to an interval $[a_i, b_i]$ on the real line, and $T_i < T_j$ if and only if $b_i < a_j$) [61]. Remains NP-complete for arbitrary K if $<$ is the disjoint union of an out-forest (or in-forest) with a layered order, or a union of layered orders, or an intersection of two layered orders [25] (a layered order is a level order with just one component).

[9] LINEAR PROGRAMMING

INSTANCE: Integer-valued vectors $V_i = (v_i[1], v_i[2], \ldots, v_i[n])$, $1 \le i \le m$, $D = (d_1, d_2, \ldots, d_m)$, and $C = (c_1, c_2, \ldots, c_n)$, and an integer B.

QUESTION: Is there a vector $X = (x_1, x_2, \ldots, x_n)$ of rational numbers such that, for $1 \le i \le m$, $V_i \cdot X \le d_i$ and such that $C \cdot X \ge B$?

Comment. This problem has been shown to be solvable in polynomial time by the ''ellipsoid method'' in the by now famous paper of L. G. Khachiyan [40] (See [4,24] for alternative presentations of the algorithm and [46] for an entertaining account of how the paper became ''famous''). Still open is the question of whether there is an algorithm for the problem which runs in polynomial time in the model of computation where inputs are arbitrary real numbers and the basic arithmetic operations all are assumed to take constant time. Such an algorithm might have more practical consequences than have so far been found for the ellipsoid method. A survey of recent research into the ellipsoid method can be found in [8]. In particular, it has been used to solve the convex quadratic programming problem in polynomial time [42] as well as the linear complementarity problem for positive definite symmetric matrices [11] (The general LINEAR COMPLEMENTARITY problem, mentioned in the comments to this problem in [G&J], has been proved NP-complete by a number of researchers [10,35]). The ellipsoid method has also been used to prove new NP-hardness results, as an equivalence between ''feasability'' and ''separability'' problems can be derived from it [28,29,39]. Finally, the furor over the ellipsoid method has revived interest in the theoretical aspects of the simplex method, which works so well in practice even though it has exponential worst case running time under the standard pivoting rules. In [15], G. B. Dantzig analyzes the *expected* running time of the simplex algorithm applied to the special case of LINEAR PROGRAMMING in which there is a convexity constraint, and shows that under reasonable assumptions it is bounded by a low-order polynomial.

[10] TOTAL UNIMODULARITY

INSTANCE: An $m \times n$ matrix M with entries from the set $\{-1, 0, 1\}$.

QUESTION: Is M ¬ totally unimodular, i.e., is there a square submatrix of M whose determinant is ¬ in the set $\{-1,0,1\}$?

Comment. This problem turns out to be solvable in polynomial time, as a consequence of an elegant characterization theorem proved by P. D. Seymour [63]. Continuing work has lowered the (high) order of the polynomial, both for the general problem [18] and for special cases [70], and used these results to derive polynomial time algorithms for integer programming when the matrix is totally unimodular [18,57,70].

[11] COMPOSITE NUMBER

INSTANCE: Positive integer N.

QUESTION: Are there positive integers $p,q > 1$ such that $N = p \cdot q$?

Comment. This problem remains open. However, further evidence that the problem is not NP-complete comes from a new algorithm [1,2,49] whose running time is $O(n^{c\, log(log(n))})$, where $n = log(N)$ is the number of bits needed to represent N. If COMPOSITE NUMBER were NP-complete, all NP-complete problems could be solved in similar running times. No similarly efficient algorithm has been found for determining the prime factors of N, and this latter problem is still widely believed to be harder than the basic decision problem.

[12] MINIMUM LENGTH TRIANGULATION

INSTANCE: Collection $C = \{(a_i,b_i): 1 \le i \le n\}$ of pairs of integers, giving the coordinates of n points in the plane, and a positive integer B.

QUESTION: Is there a triangulation of the set of points represented by C that has total ''discrete-Euclidean'' length B or less? Here a triangulation is a collection of non-intersecting line segments, each joining two points in C, that divides the interior of the convex hull into triangular regions. The discrete-Euclidean length of a line segment joining (a_i,b_i) and (a_j,b_j) is given by $\lceil ((a_i - a_j)^2 + (b_i - b_j)^2)^{1/2} \rceil$, and the total length of a triangulation is the sum of the lengths of its constituent line segments.

Comment. Polynomial-time algorithms that were once conjectured to solve this problem, but were already known to be nonoptimal when [G&J] was prepared, have now been shown to do so poorly on some instances that they may construct a triangulation whose total length is an arbitrary multiple of the optimal length [41,56].

I shall conclude this first column by mentioning two more ''open'' problems (which should have been on our original list, but weren't), and by pointing out a number of minor errors that have been found in the list of NP-complete problems in [G&J]. The first of the two problems has been the subject of much recent interest, and, like so many of the problems on our original list, can now only be called ''previously open.'' It asks about the complexity of the INTEGER PROGRAMMING problem [MP1], when restricted to a fixed number K of variables. The problem is trivial for $K = 1$, and the special case of $K = 2$ when all coefficients are non-negative was recently shown to be solvable in polynomial time [36]. However, the general case for $K \geq 2$ remained open until this year. In [48], H. W. Lenstra uses ideas from the ''geometry of numbers'' to develop a polynomial time algorithm for each fixed value of K. (Unfortunately, as with all such sequences of algorithms mentioned here, the running times are exponential in K). Lenstra also obtains, as a corollary, that INTEGER PROGRAMMING is solvable in polynomial time for any fixed number of constraints, so long as the solution vector can contain arbitrary integers. (If the solution vector is required to be non-negative, INTEGER PROGRAMMING is NP-complete even for two constraints, but can be solved in pseudo-polynomial time for any fixed number of constraints [60]).

The second problem concerns perfect graphs, and I am happy to say that it is still open (as of this writing) and hence can be offered as a partial replacement for the six problems already eliminated from our list. It was omitted from the original list because of a technical objection (it was not known to be in NP) which has recently been removed.

[OPEN] IMPERFECT GRAPH

INSTANCE: Graph $G = (V,E)$.

QUESTION: Is G *not* a perfect graph, i.e., is there a subset $V' \subseteq V$ such that the subgraph of G induced by V' has a chromatic number which is larger than its maximum clique size?

Comment. Perfect graphs are of wide interest in combinatorial theory - entire books have been written about them (e.g., see [27]). That this problem is in NP would follow immediately from the famous ''strong perfect graph conjecture,'' which says that a graph is imperfect if and only if its complement contains an induced subgraph which is an odd cycle of length 5 or more. However, Grötschel, Lovász, and Schrijver [29] have recently shown that the problem is in NP without proving this conjecture (instead, they use the ellipsoid method). They can also show by related techniques that weighted versions of the CHROMATIC NUMBER, CLIQUE, CLIQUE COVER, and INDEPENDENT SET problems are all solvable in polynomial time when restricted to perfect graphs [28,29] (note that these problems are all equivalent for perfect graphs, given the ''perfect graph theorem'' which says that the complement of a perfect graph is perfect). The algorithms are applicable to all graphs, and either report that the graph is not perfect, or return the correct answer (in which case the graph may or may not be perfect).

As to corrections to [G&J]: Despite our claims to the contrary, HAMILTONIAN CIRCUIT [GT37] is NP-complete for edge graphs. Our claim of polynomial time solvability was based on a faulty analogy with Euler tours, as was pointed out to us by D. Skrien. MINIMUM TEST SET [SP6] is NP-complete even when restricted to the case where each subset has at most two elements, as was pointed out to us by J. K. Lenstra. SUBSET PRODUCT [SP14], although not solvable in time polynomial in $|A|$ and $\max\{s(a)\colon a \in A\}$, is solvable in pseudo-polynomial time due to the presence of B in the input, as was pointed out to us by T. Ibaraki. Finally, in our comments on QUADRATIC DIOPHANTINE EQUATIONS [AN8], our claim that $\sum_{i=1}^{k} a_i x_i = c$ is solvable in polynomial time holds only if arbitrary integer solutions are allowed. If only non-negative solutions are allowed, the problem is of course NP-complete. This imprecision was pointed out to us by J. Shepherdson.

REFERENCES

1. L. M. ADLEMAN, On distinguishing prime numbers from composite numbers, *in* ''Proceedings 21st Ann. Symp. on Foundations of Computer Science,'' pp. 387-406, IEEE Computer Society, Los Angeles, 1980.
2. L. M. ADLEMAN, C. POMERANCE, AND R. S. RUMELY, On distinguishing prime numbers from composite numbers, manuscript (1980).
3. A. V. AHO, J. E. HOPCROFT AND J. D. ULLMAN, ''The Design and Analysis of Computer Algorithms,'' Addison-Wesley, Reading, MA, Chapter 10, 1974.
4. B. ASPVALL AND R. E. STONE, Khachiyan's linear programming algorithm, *J. Algorithms* **1** (1980), 1-13.
5. L. BABAI, Monte-Carlo algorithms in graph isomorphism testing, manuscript (1979).
6. L. BABAI, Moderately exponential bound for graph isomorphism, ''Proc. Conf. on Fundamentals of Computation Theory,'' pp. to appear., Lecture Notes in Computer Science, Springer, Berlin, 1981.
7. L. BABAI AND L. KUCERA, Graph canonization in linear average time, *in* ''Proceedings 20th Ann. Symp. on Foundations of Computer Science,'' pp. 39-46, IEEE Computer Society, Los Angeles, 1979.
8. R. G. BLAND, D. GOLDFARB, AND M. J. TODD, The ellipsoid method: A survey, Report No. 476, School of Operations Research and Industrial Engineering, Cornell University, Ithaca, NY.
9. K. S. BOOTH AND C. J. COLBOURN, Problems polynomially equivalent to graph isomorphism, Report No. CS-77-04, Dept. of Computer Science, University of Toronto, Waterloo, Ontario.
10. S. J. CHUNG, A note on the complexity of LCP: the LCP is NP-complete, Report No. 79-2, Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor.
11. S. J. CHUNG AND K. G. MURTY, A polynomial bounded algorithm for positive definite symmetric LCPs, Report No. 79-10, Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor.
12. C. J. COLBOURN, Isomorphism complete problems on matrices, ''Proc. West Coast Conf. on Combinatorics, Graph Theory, and Computing,'' pp. 101-107., Congressus Numerantium XXVI, Utilitas Mathematica Publishing, Winnipeg, 1980.

13. M. J. COLBOURN AND C. J. COLBOURN, Concerning the complexity of deciding isomorphism of block designs, *Discrete Applied Math.* **3** (1981), 155-162.
14. C. J. COLBOURN AND D. G. CORNEIL, On deciding switching equivalence of graphs, *Discrete Applied Math.* **2** (1980), 181-184.
15. G. B. DANTZIG, Expected number of steps of the simplex method for a linear program with a convexity constraint, Report No. SOL 80-3R, Systems Optimization Laboratory, Department of Operations Research, Stanford University, Stanford, CA.
16. D. DOLEV, Scheduling wide graphs, Report No. STAN-CS-80-832, Department of Computer Science, Stanford University, Stanford, CA.
17. B. DUSHNIK, AND E. W. MILLER, Partially ordered sets, *Amer. J. Math.* **63** (1941), 600-610.
18. J. EDMONDS, Seymour's theorem and good algorithms for totally unimodular matrices, in preparation.
19. I. S. FILOTTI, An algorithm for embedding cubic graphs in the torus, *J. Comput. System Sci.* **20** (1980), 255-276.
20. I. S. FILOTTI AND J. N. MAYER, A polynomial-time algorithm for determining the isomorphism of graphs of fixed genus, *in* ''Proceedings 12th Ann. ACM Symp. on Theory of Computing,'' pp. 236-243, Association for Computing Machinery, New York, 1980.
21. I. S. FILOTTI, G. L. MILLER, AND J. REIF, On determining the genus of a graph in $O(V^{O(g)})$ steps, *in* ''Proceedings 11th Ann. ACM Symp. on Theory of Computing,'' pp. 27-37, Association for Computing Machinery, New York, 1979.
22. S. FORTUNE, J. HOPCROFT, AND J. WYLLIE, The directed subgraph homeomorphism problem, *Theor. Comput. Sci.* **10** (1980), 111-121.
23. M. FURST, J. HOPCROFT, AND E. LUKS, Polynomial-time algorithms for permutation groups, *in* ''Proceedings 21st Ann. Symp. on Foundations of Computer Science,'' pp. 36-41, IEEE Computer Society, Los Angeles, 1980.
24. P. GÁCS AND L. LOVÁSZ, Khachian's algorithm for linear programming, *Math. Prog. Studies* **14** (1981), 61-68.
25. M. R. GAREY, D. S. JOHNSON, R. E. TARJAN, AND M. YANNAKAKIS, Scheduling opposing forests, manuscript (1980).
26. M. K. GOLDBERG, A nonfactorial algorithm for testing isomorphism of two graphs, manuscript (1981).
27. M. C. GOLUMBIC, ''Algorithmic Graph Theory and Perfect Graphs,'' Academic Press, New York, 1980.
28. M. GRÖTSCHEL, L. LOVÁSZ, AND A. SCHRIJVER, The ellipsoid method and its consequences in combinatorial optimization, *Combinatorica* **1** (1981), 169-197.
29. M. GRÖTSCHEL, L. LOVÁSZ, AND A. SCHRIJVER, Polynomial algorithms for perfect graphs, manuscript (1981).
30. T. HIRATU AND M. KIMURA, The subgraph homeomorphism problem on reducible flowgraphs, ''Graph Theory and Algorithms,'' pp. 69-84., Proc. of 17th Symp. of Research Institute of Electrical Communication, Tohoku University, Sendai, Japan, 1980.
31. C. HOFFMANN, Testing isomorphisms of cone graphs, *in* ''Proceedings 12th Ann. ACM Symp. on Theory of Computing,'' pp. 244-251, Association for Computing Machinery, New York, 1980.
32. I. HOLYER, The NP-completeness of edge coloring, *SIAM J. Comput.*, to appear.
33. E. HOROWITZ, AND S. SAHNI, ''Algorithms: Design and Analysis,'' Computer Science Press, Potomac, MD, Chapter 11, 1978.
34. T. C. HU, Parallel sequencing and assembly line problems, *Operations Res.* **9** (1961), 841-848.
35. G. HUBERMAN, private communication (1979).
36. R. KANNAN, A polynomial algorithm for the two-variable integer programming problem, *J. Assoc. Comput. Mach.* **27** (1980), 118-122.

37. R. M. KARP, Reducibility among combinatorial problems, in R. E. Miller and J. W. Thatcher (eds.), ''Complexity of Computer Computations,'' pp. 85-103., Plenum Press, New York, 1972.
38. R. M. KARP, On the complexity of combinatorial problems, *Networks* **5** (1975), 45-68.
39. R. M. KARP AND C. H. PAPADIMITRIOU, On linear characterizations of combinatorial optimization problems, *in* ''Proceedings 21st Ann. Symp. on Foundations of Computer Science,'' pp. 1-9, IEEE Computer Society, Los Angeles, 1980.
40. L. G. KHACHIYAN, A polynomial algorithm in linear programming, *Dokl. Akad. Nauk. SSSR* **244** (1979), 1093-1096 (in Russian).. English translation in *Soviet Math. Dokl.* **20** (1979), 191-194.
41. D. G. KIRKPATRICK, A note on Delaunay and optimal triangulations, *Inform. Process. Lett.* **10** (1980), 127-128.
42. M. K. KOZLOV, S. P. TARASOV, AND L. G. KHACHIYAN, Polynomial solvability of convex quadratic programming, *Doklady Akademiia Nauk USSR* **248** (1979), 1049-1051 (in Russian).. English translation in *Soviet Math. Dokl.*, **20**, 1979.
43. B. J. LAGEWEG, E. L. LAWLER, J. K. LENSTRA, AND A. H. G. RINNOOY KAN, Computer aided complexity classification of deterministic scheduling problems, Report No. BW 138/81, Stichting Mathematisch Centrum, Amsterdam.
44. A. S. LAPAUGH AND R. L. RIVEST, The subgraph homeomorphism problem, *J. Comput. System Sci.* **20** (1980), 133-149.
45. E. L. LAWLER, ''Combinatorial Optimization: Networks and Matroids,'' Holt, Rinehart and Winston, New York, 1976.
46. E. L. LAWLER, The great mathematical Sputnik of 1979, *The Sciences*, September 1980, 12-15,34-35.
47. E. L. LAWLER, AND O. VORNBERGER, The partial order dimension problem is NP-complete, manuscript (1981).
48. H. W. LENSTRA, JR, Integer programming with a fixed number of variables, Report No. 81-03, Department of Mathematics, University of Amsterdam, Amsterdam.
49. H. W. LENSTRA, JR, Primality testing algorithms (after Adleman, Rumely and Williams), Séminaire BOURBAKI 1980/1981 No. 576.
50. D. LEVEN, AND Z. GALIL, NP-complete problem number 798016, manuscript (1981).
51. D. LICHTENSTEIN, Isomorphism for graphs embeddable on the projective plane, *in* ''Proceedings 12th Ann. ACM Symp. on Theory of Computing,'' pp. 218-224, Association for Computing Machinery, New York, 1980.
52. P. C. LIU AND R. C. GELDMACHER, An $O(max(m,n))$ algorithm for finding a subgraph homeomorphic to K_4, *Congressus Numerantium* **29** (1980), 597-609.
53. L. LOVÁSZ, The matroid matching problem, *Proc. Conf. Algebraic Methods in Graph Theory (Szeged, 1978)*, (to appear).
54. L. LOVÁSZ, Matroid matching and some applications, *J. Combinatorial Theory Ser. B* **28** (1980), 208-236.
55. E. M. LUKS, Isomorphism of bounded valence can be tested in polynomial time, *in* ''Proceedings 21st Ann. Symp. on Foundations of Computer Science,'' pp. 42-49, IEEE Computer Society, Los Angeles, 1980.
56. G. K. MANACHER AND A. L. ZOBRIST, Neither the greedy nor the Delaunay triangulation of a planar set approximates the optimal triangulation, *Inform. Process. Lett.* **9** (1979), 31-34.
57. J. F. MAURRAS, K. TRUEMPER, AND M. AKGUL, Polynomial algorithms for a class of linear programs, *Math. Programming* **21** (1981), 121-136. manuscript (1981).
58. G. L. MILLER, Isomorphism testing for graphs of bounded genus, *in* ''Proceedings 12th Ann. ACM Symp. on Theory of Computing,'' pp. 225-235, Association for Computing Machinery, New York, 1980.
59. T. OHTSUKI, The two disjoint path problem and wire routing design, ''Graph Theory and Algorithms,'' pp. 257-267., Proc. of 17th Symp. of Research Institute of Electrical Communication, Tohoku University, Sendai, Japan, 1980.

60. C. H. PAPADIMITRIOU, On the complexity of integer programming, *J. Assoc. Comput. Mach.*, to appear.
61. C. H. PAPADIMITRIOU AND M. YANNAKAKIS, Scheduling interval-ordered tasks, *SIAM J. Comput.* **8** (1979), 405-409.
62. A. PNUELI, A. LEMPEL, AND S. EVEN, Transitive orientation of graphs and identification of permutation graphs, *Canad. J. Math.* **23** (1971), 160-175.
63. P. D. SEYMOUR, Decomposition of regular matroids, *J. Combinatorial Theory Ser. B* **28** (1980), 305-359.
64. P. D. SEYMOUR, Disjoint paths in graphs, *Discrete Math.* **29** (1980), 293-309.
65. Y. SHILOACH, A polynomial solution to the undirected two paths problem, *J. Assoc. Comput. Mach.* **27** (1980), 445-456.
66. J. VALDES, Subtree homeomorphism and tree contractability, manuscript (1980).
67. M. WARMUTH, private communication (1981).
68. M. YANNAKAKIS, Computing the minimum fill-in is NP-complete, *SIAM J. Algebraic and Discrete Methods* **2** (1981), 77-79.
69. M. YANNAKAKIS, The complexity of the partial order dimension problem, manuscript (1981).
70. M. YANNAKAKIS, On a class of totally unimodular matrices, *in* ''Proceedings 21st Ann. Symp. on Foundations of Computer Science,'' pp. 10-16, IEEE Computer Society, Los Angeles, 1981.
71. V. N. ZEMLYACHENKO, in preparation (1981).