Discrete Mathematics 126 (1994) 349-357
North-Holland

Deciding whether a planar graph has a cubic subgraph is NP-complete

Iain A. Stewart

Computing Laboratory, University of Newcastle upon Tyne, Claremont Tower, Claremont Road, Newcastle upon Tyne, NE1 7RU, UK

Received 26 September 1991
Revised 30 April 1992

Abstract

We show that the problem of deciding whether a given planar graph (complete with planar embedding) of degree at most 7 has a cubic subgraph is NP-complete.

In Garey and Johnson's compendium of NP-complete problems [1], many of the references are personal communications between themselves and others, and the proofs of some NP-completeness results have not appeared before in print. One such (to our knowledge) is the proof that CUBIC SUBGRAPH is NP-complete, attributed to Chvàtal in [1], where CUBIC SUBGRAPH is the problem of deciding whether a given graph has a nontrivial subgraph with each vertex of degree 3, i.e. a cubic subgraph: the hint given in [1] is to reduce GRAPH 3-COLOURABILITY to CUBIC SUBGRAPH. The nonexistence of such proofs in the literature has probably restricted further interest in these and related problems.

In this paper we exhibit a logspace reduction from ONE-IN-THREE 3SAT to CUBIC SUBGRAPH and then show how, from this reduction, a logspace reduction from ONE-IN-THREE 3SAT to CUBIC SUBGRAPH PLANAR can be constructed: ONE-IN-THREE 3SAT is the version of 3-SATISFIABILITY where exactly one literal in each clause of exactly 3 literals is satisfied and it was shown to be complete for NP via logspace reductions in [3], and CUBIC SUBGRAPH PLANAR is the version of CUBIC SUBGRAPH where all instances are planar graphs. In fact, we show that the problem obtained from CUBIC SUBGRAPH PLANAR where all

Correspondence to: Iain A. Stewart, Department of Computer Science, University College of Swansea, Singleton Park, Swansea SA2 8PP, UK.

SSDI 0012-365X(92)00435-Z

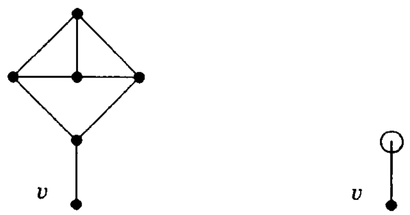

Fig. 1. A tag on a vertex v and its pictoral abbreviation.

instances are of degree at most 7 and come with a planar embedding is complete for NP via logspace reductions. Our results provide more information on those properties which are hard to verify computationally for both general and planar graphs.

Our technique is similar to that used in [2], although the word 'technique' is probably a misnomer. What we do is to convert the logspace reduction from ONE-IN-THREE 3SAT to CUBIC SUBGRAPH into one from ONE-IN-THREE 3SAT to CUBIC SUBGRAPH PLANAR by replacing any crossing edges with a suitable planar graph. As remarked in [2], it is often feasible to use this approach when specializing a given reduction to planar graphs, but finding appropriate graphs with which to replace crossing edges is usually hard.

Before we proceed with the proof of Theorem 1, we need to introduce some nonstandard notation. We say that there is a *tag* on a vertex v of a graph G if there is a subgraph of G as in Fig. 1 such that no other edges of G involve the vertices of the tag (except for perhaps v): we abbreviate a tag pictorially as in Fig. 1. A vertex can have any number of tags on it in some graph. Notice that one edge of a tag occurs in a cubic subgraph of G if and only if all of the edges of the tag do.

Theorem 1. *There exists a logspace reduction from* ONE-IN-THREE 3SAT *to* CUBIC SUBGRAPH.

Proof. Let (X, C) be an instance of ONE-IN-THREE 3SAT, where $X=\{X_1, X_2, \ldots, X_p\}$ is a set of Boolean variables and $C=\{C_1, C_2, \ldots, C_q\}$ is a set of clauses, each containing exactly 3 literals. We build the corresponding instance G of CUBIC SUBGRAPH in stages.

Stage A: For each $i \in \{1, 2, \ldots, p\}$, we build the graph G_i as follows.

(i) G_i has vertices:

$$\{X_i, \neg X_i, X_{i,j}, \neg X_{i,j}, Y_i, \neg Y_i, Y_{i,k}, \neg Y_{i,k}, U_{i,k}, \neg U_{i,k}, V_{i,k}, \neg V_{i,k}\colon j=1,2,3;\ k=1,2,\ldots,q\},$$

amongst others.

(ii) G_i has edges:

$$S_1=\{(X_i, X_{i,1}), (X_{i,1}, X_{i,2}), (X_{i,2}, X_{i,3}), (X_{i,3}, Y_i), (X_{i,j}, \neg X_i)\colon j=1,2,3\};$$

$$S_2 = \{\text{two tags on } X_i, \text{ one tag on } Y_i\};$$

$$\neg S_1 = \{(\neg X_i, \neg X_{i,1}), (\neg X_{i,1}, \neg X_{i,2}), (\neg X_{i,2}, \neg X_{i,3}), (\neg X_{i,3}, \neg Y_i), (\neg X_{i,j}, X_i)\colon j = 1, 2, 3\};$$

$$\neg S_2 = \{\text{two tags on } \neg X_i, \text{ one tag on } \neg Y_i\}.$$

(iii) If X_i (resp. $\neg X_i$) appears in the clauses $C_{i_1}, C_{i_2}, \ldots, C_{i_r}$, for some $r \geqslant 1$, then G_i has edges:

$$S_3 = \{(Y_{i,i_j}, Y_{i,i_{j+1}}), (Y_{i,i_j}, V_{i,i_j})\colon j = 1, 2, \ldots, r-2\} \cup \{(Y_{i,i_{r-1}}, V_{i,i_r}), (Y_i, Y_{i,i_1})\} \cup \{(U_{i,i_j}, V_{i,i_j})\colon j = 1, 2, \ldots, r\} \cup \{\text{one tag on } U_{i,i_j}\colon j = 1, 2, \ldots, r\}$$

(resp.

$$\neg S_3 = \{(\neg Y_{i,i_j}, \neg Y_{i,i_{j+1}}), (\neg Y_{i,i_j}, \neg V_{i,i_j})\colon j = 1, 2, \ldots, r-2\} \cup \{(\neg Y_{i,i_{r-1}}, \neg V_{i,i_r}), (\neg Y_i, \neg Y_{i,i_1})\} \cup \{(\neg U_{i,i_j}, \neg V_{i,i_j})\colon j = 1, 2, \ldots, r\} \cup \{\text{one tag on } \neg U_{i,i_j}\colon j = 1, 2, \ldots, r\}).$$

Stage B: The graph G_0 is defined as follows.
(i) G_0 has vertices:

$$\{C_j, D_j\colon j = 1, 2, \ldots, q\},$$

amongst others.
(ii) G_0 has edges:

$$\{(C_j, D_j)\colon j = 1, 2, \ldots, q\} \cup \{(D_j, D_{j+1})\colon j = 1, 2, \ldots, q-1\} \cup \{\text{one tag on } D_1, \text{ one tag on } D_q \text{ (or two tags on } D_1 \text{ if } q = 1)\}.$$

Stage C: The graph G is the (disjoint) union of the graphs $\{G_i\colon i = 0, 1, \ldots, p\}$, together with the edges:

$$T_i = \{(U_{i,j}, C_j), (V_{i,j}, C_j)\colon j = 1, 2, \ldots, q;\ X_i \in C_j\},$$

$$\neg T_i = \{(\neg U_{i,j}, C_j), (\neg V_{i,j}, C_j)\colon j = 1, 2, \ldots, q;\ \neg X_i \in C_j\},$$

for $i = 1, 2, \ldots, p$.

The subgraph of G consisting of G_0, some G_i, for $i \in \{1, 2, \ldots, p\}$, and the edges between G_0 and G_i is shown in Fig. 2: we assume in this particular instance that X_i (resp. $\neg X_i$) appears in the clauses C_1, C_3 and C_4 (resp. C_1, C_2 and C_3). For each $i = 1, 2, \ldots, p$, we call the edges of $S_1 \cup S_2 \cup S_3 \cup T_i$ (resp. $\neg S_1 \cup \neg S_2 \cup \neg S_3 \cup \neg T_i$) the edges *belonging to* X_i (resp. $\neg X_i$).

Suppose that (X, C) is a yes-instance of ONE-IN-THREE 3SAT. So, there is a truth assignment t on the Boolean variables of X such that exactly one literal in each clause of C is set at *True* under t. Consider the graph G corresponding to (X, C), as defined above. In G, mark all edges of G_0, and for each $i \in \{1, 2, \ldots, p\}$, if X_i (resp. $\neg X_i$) is set at

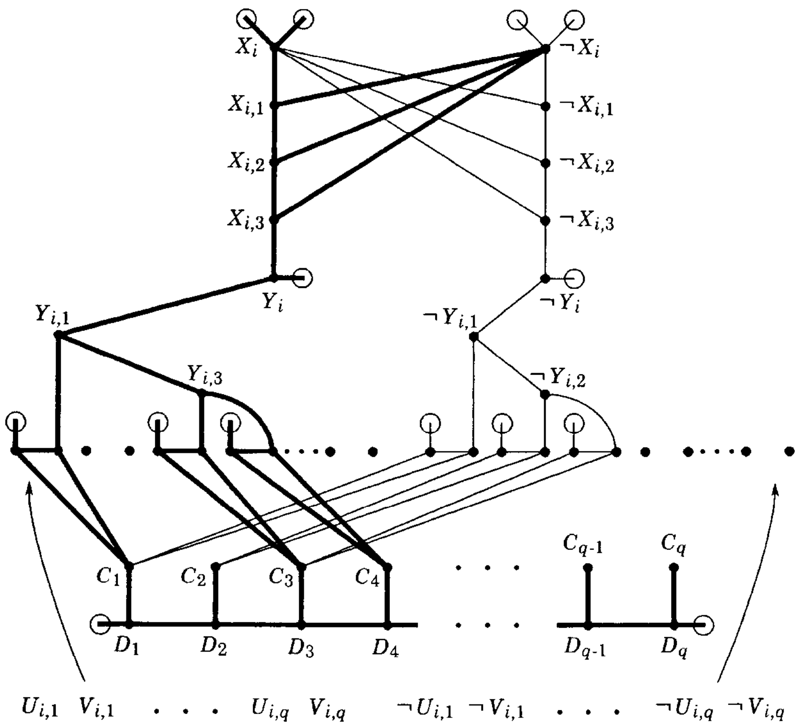

Fig. 2. The subgraph induced by G_0 and G_i.

True under t and appears in some clause of C then mark all those edges of G belonging to X_i (resp. $\neg X_i$). In the illustration of the subgraph of G consisting of G_0, G_i, for some $i \in \{1, 2, \ldots, p\}$, and the edges between G_0 and G_i, in Fig. 2, the edges belonging to X_i and the edges of G_0 are drawn in bold. Let E be the set of marked edges of G and let H be the subgraph of G with edges E. As the edges belonging to X_i and $\neg X_i$, for some i, cannot be simultaneously marked, the only way that H could fail to be cubic is for some vertex C_i to have degree different from 3 in H. This is impossible as every clause of C has exactly one literal set at *True* under t. Hence, G has a cubic subgraph.

Conversely, suppose that G has a cubic subgraph H, where G is the graph corresponding to some instance (X, C) of ONE-IN-THREE 3SAT. It is easy to see that all edges of G_0 must appear in H, and consequently that each vertex C_i must have degree 3 in H. For each C_i, $i \in \{1, 2, \ldots, q\}$, let L_i be the unique literal for which two of the edges belonging to L_i involve C_i, and set $L = \{L_1, L_2, \ldots, L_q\}$. It should be clear that X_i and $\neg X_i$, for some i, cannot both be in L (as, for instance, there is no cubic subgraph of G_i whose edges involve both the vertices Y_i and $\neg Y_i$). Define the truth assignment t by setting $t(L_i) = True$, for each $i = 1, 2, \ldots, q$. This truth assignment is well-defined and is such that exactly one literal in each clause C_i is set at *True* under t (as each vertex C_i has degree 3 in H). Hence, (X, C) is a yes-instance.

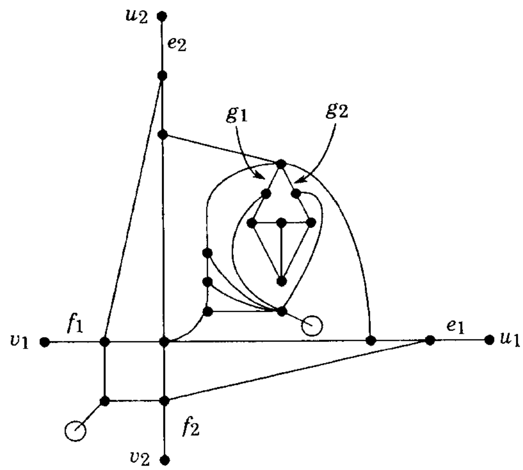

Fig. 3. The graph K of Proposition 3.

Given an instance (X, C) of ONE-IN-THREE 3SAT, the corresponding instance G of CUBIC SUBGRAPH can clearly be constructed in logspace and so the result follows. □

Schaefer's result, mentioned earlier, now implies that of Chvàtal, also mentioned earlier.

Corollary 2. CUBIC SUBGRAPH *is complete for* NP *via logspace reductions.*

The following proposition involves the graph we are to use to replace crossing edges in a nonplanar graph (see the initial preamble).

Proposition 3. *Consider the graph K in Fig.* 3. *If K is a subgraph of some graph G such that no other edges of G involve the vertices of* $K \backslash \{u_1, u_2, v_1, v_2\}$ *and H is a cubic subgraph of G, then the edge* e_i *appears in H if and only if the edge* f_i *appears in H, for* $i = 1, 2$.

Proof (*sketch*). In turn, consider the cases when e_1 and e_2 both appear in H, when one of them appears in H, and when none of them appears in H. For each case, mark the rest of the edges of K remembering that each vertex of $\mathrm{K} \backslash \{u_1, u_2, v_1, v_2\}$ must be of degree 3 or 0 in H: notice that either g_1 and g_2 are both in H or neither of them is in H. □

Theorem 4. *There exists a logspace reduction from* ONE-IN-THREE 3SAT *to* CUBIC SUBGRAPH PLANAR.

Proof. Consider the logspace reduction from ONE-IN-THREE 3SAT to CUBIC SUBGRAPH as described in the proof of Theorem 1 (we work throughout with the terminology introduced in the proof of Theorem 1). It should be clear that the graph G corresponding to some instance (X, C) of ONE-IN-THREE 3SAT is planar except, possibly, for some crossing edges joining the subgraphs $\{G_i: i=1,2,\dots,p\}$ to the subgraph G_0 (any G_i is planar even though it is not drawn as such in Fig. 2). Consequently, if we can 'remove' these crossings (by introducing more vertices and edges) whilst still retaining the pertinent properties of G (and do this in logspace), then we are done.

Our strategy is as follows. Firstly, we show that an output graph G from the logspace reduction from ONE-IN-THREE 3SAT to CUBIC SUBGRAPH, mentioned above, can be transformed, in logspace, to another graph G' where the crossing edges of G are encoded as line segments on a grid, but where G and G' are, in fact, identical. Secondly, we show that a graph G' can be transformed, in logspace, to another graph G'' (essentially by adding some extra vertices and edges) such that G' has a cubic subgraph if and only if G'' has. Finally, we show that the crossings of the graph G'' (which are essentially the crossings of the graph G) can be removed by replacing each crossing with a fixed planar graph in the style of [2]. The grid embedding of the relevant subraph of G'' (which is still available) enables us to systematically identify and remove each crossing whilst using only logspace.

Consider the subgraph M of such a graph G, above, consisting of all edges with one end-vertex in $\{U_{i,j}, V_{i,j}, \neg U_{i,j}, \neg V_{i,j}: i=1,2,\dots,p; j=1,2,\dots,q\}$ and the other in $\{C_j: j=1,2,\dots,q\}$. For ease of readability, we prefer to rename the vertices of $\{U_{i,j}, V_{i,j}, \neg U_{i,j}, \neg V_{i,j}: i=1,2,\dots,p; j=1,2,\dots,q\}$ as follows:

for $i=1,2,\dots,p$ and $j=1,2,\dots,4q=r$, $w_{i,j}$ is defined as

$$\begin{array}{llll} U_{i,j+1/2} & \text{if } j\leqslant 2q \text{ and odd;} & V_{i,j/2} & \text{if } j\leqslant 2q \text{ and even;} \\ \neg U_{i,j+1-2q/2} & \text{if } j>2q \text{ and odd;} & \neg V_{i,j-2q/2} & \text{if } j>2q \text{ and even.} \end{array}$$

Also, for each C_j we introduce new vertices $\{z_{j,i}: i=1,2,\dots,r\}$.

We rearrange the vertices of $\{w_{i,k}, z_{j,k}: i=1,2,\dots,p;\ k=1,2,\dots,r; j=1,2,\dots,q\}$ on the perimeter of a grid as shown in Fig. 4. If there is an edge $(w_{i,k}, C_j)$ in G then we draw a horizontal line segment on the grid from $w_{i,k}$ to the node of the grid vertically above $z_{j,k}$, and then draw a vertical line segment down to $z_{j,k}$. The grid, complete with line segments corresponding to the edges of the subgraph M of the graph G of Fig. 2, is shown in Fig. 4 (with the line segments drawn in bold and assuming that the variable i in Fig. 2 is actually equal to 1). By removing each vertex $z_{j,k}$ from the grid and extending any vertical line segment involving $z_{j,k}$ to the vertex C_j, we clearly have another representation of the subgraph M of G. In particular, the nodes of the grid where line segments cross correspond to the crossing points of the edges of M in the original graph G. Denoting our new representation of G, where the edges of M are represented as line segments on a grid, by G', it is easy to see how the graph G' can be

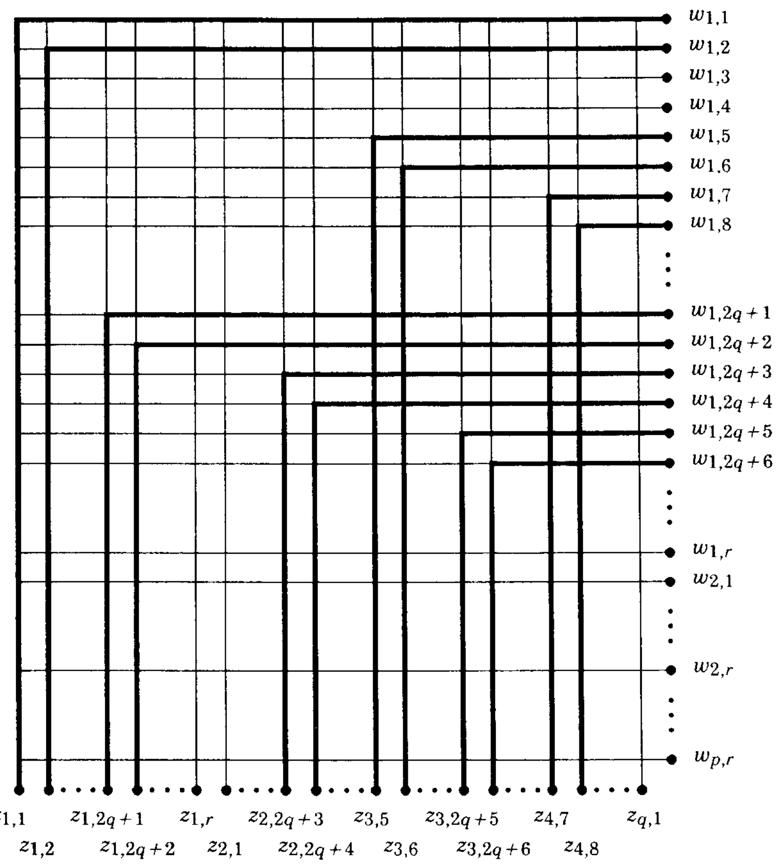

Fig. 4. The grid and some line segments.

obtained from G in logspace (G and G' are actually identical, but more information is present in the encoding of G' regarding the embedding of M on the grid).

Having completed the first phase of our strategy, we now amend the graph G' slightly. Consider the embedding of M on the grid in G', as described above. For each crossing point of line segments on the grid, we place a tagged vertex to the left of (resp. to the right of, above, below) the crossing point on the horizontal (resp. horizontal, vertical, vertical) line segment involved in the crossing: these tagged vertices should appear before any other introduced vertices or crossing points. A portion of the amended grid of Fig. 4 is shown in Fig. 5 (in order to refer to the line segments on the grid, we pretend that the vertices of $\{z_{j,k}: j=1,2,\ldots,q;\ k=1,2,\ldots,r\}$ are still present even though in reality they are not). Denoting the amended graph by G'', it is easy to see that G'' can be constructed from G' in logspace (we retain the grid structure in the encoding of G''), and that G' has a cubic subgraph if and only if G'' does. This completes the second phase of our strategy.

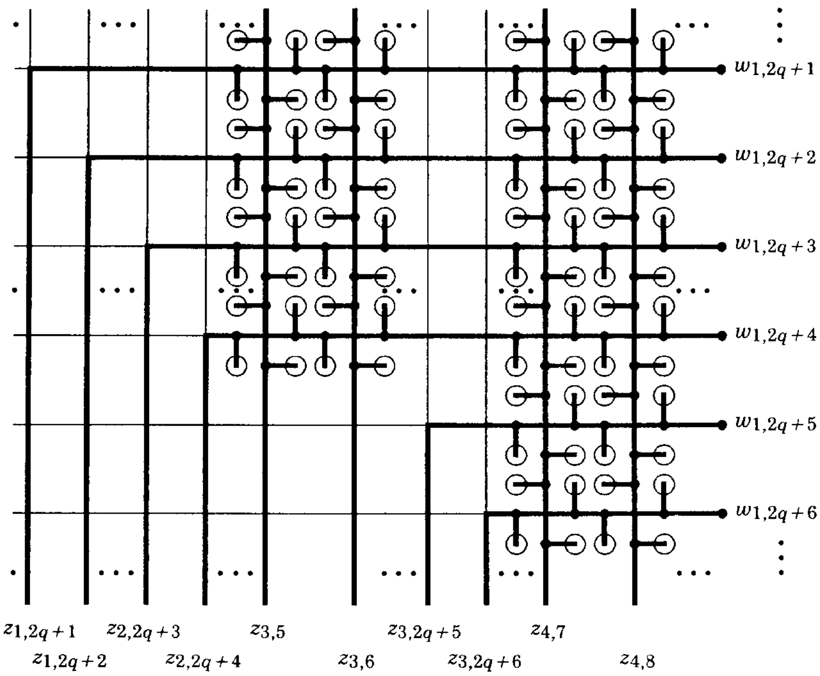

Fig. 5. The amended graph G''.

Finally, we must remove the crossings of G'' (in logspace: remember that G'' is planar except possibly for the grid crossings of the line segments). Notice that the addition of the vertices and edges to G', to get G'', has increased the number of rows and columns of our grid: this is of no consequence. Consider a crossing point in the grid. Then according to the construction in the second phase, 4 new tagged vertices have been introduced: l, r, a and b, where l (resp. r, a, b) is the vertex to the left of (resp. to the right of, above, below) the crossing point. Remove the edges (l, r) and (a, b) from the graph and replace them with the graph K of Proposition 3, with $u_1 = r$, $u_2 = a$, $v_1 = l$ and $v_2 = b$. By Proposition 3, the amended graph has a cubic subgraph if and only if G'' does, and one of the 'obstructions to planarity' has been removed. If we repeat this procedure for every crossing point then we obtain a planar graph G''' which has a cubic subgraph if and only if G does. The removal of all crossing points can be achieved in logspace (as there are only polynomially many of them and we can identify them using the grid structure). Consequently, as logspace reductions are transitive, there is clearly a logspace reduction from ONE-IN-THREE 3SAT to CUBIC SUBGRAPH PLANAR. □

Again, Schaefer's result yields the following corollary.

Corollary 5. CUBIC SUBGRAPH PLANAR *is complete for* NP *via logspace reductions.*

In fact, by scrutinizing the graphs involved in the proofs of Theorems 1 and 4 and Proposition 3, it is easy to see that more can be said.

Corollary 6. *The version of* CUBIC SUBGRAPH PLANAR *where all instances have degree at most* 7 *and come with a planar embedding is complete for* NP *via logspace reductions.*

References

[1] M.R. Garey and D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-completeness (Freeman, San Francisco, 1979).
[2] D. Lichtenstein, Planar formulae and their uses, SIAM J. Comput. (1982) 329–343.
[3] T.J. Schaefer, The complexity of satisfiability problems, in: Proc. 10th ACM Conf. on Theory of Computing (ACM, New York, 1978) 216–226.