A CORRECTION TO BRELAZ'S MODIFICATION OF BROWN'S COLORING ALGORITHM

JÜRGEN PEEMÖLLER
Hochschule der Bundeswehr Hamburg, Hamburg, West Germany

ABSTRACT: Brelaz's modification of Brown's exact coloring algorithm contains two errors as demonstrated in two examples. A correct version of the algorithm and a proof of the exactness are given. Finally, Brown's look-ahead rule is built into this algorithm.

Jürgen Peemöller's current projects involve graph coloring, timetabling, and Hamiltonian graphs, and he is a member of the German Society for Operations Research.

Author's Present Address: Jürgen Peemöller, Institut für Unternehmensforschung und Informatik, Hochschule der Bundeswehr Hamburg, Holstenhofweg 85, D-2000 Hamburg 70, West Germany.

INTRODUCTION

Brelaz [1] gives a modification of Brown's exact coloring algorithm (see [2]) which includes two errors. The first one occurs in Step 8 of the Brelaz algorithm, which gives the rule for removing labels. The second one occurs in step 7, which gives the rule for going back if you have found a new solution. Each of these errors can affect the elimination of an optimal solution as the following examples will show.

Example 1

Consider the graph shown in Figure 1. Let the coloration order be $x_1, \ldots, x_9$. Assume x_1, x_2, x_3 to be the initial clique and 4 to be an upper bound for the chromatic number derived from a heuristic.

The Brelaz algorithm now colors the vertices $x_1, \ldots, x_8$ with 1, 2, 3, 1, 1, 2, 3, 3. All the colors 1, 2, 3 are now prohibited for x_9. Therefore x_4 and x_7 are labeled with the label 9 (x_2 belongs to the initial clique and will be unlabeled). The backtrack goes to x_7. The only color available for this vertex is 3. So x_5 and x_6 are labeled with the label 7. The following backtrack leads to x_6 which gets the new color 3. Now you can color x_7 with 2, and (this is the error) you have to remove label 7 from x_5 and x_6. With the actual partial solution 1, 2, 3, 1, 1, 3, 2 there is no color available for x_8. The backtrack now goes from x_8 to x_6 and further to x_4, where the algorithm stops with the initial coloration of four colors. The exact solution 1, 2, 3, 1, 2, 2, 1, 3, 3 has been eliminated by the algorithm.

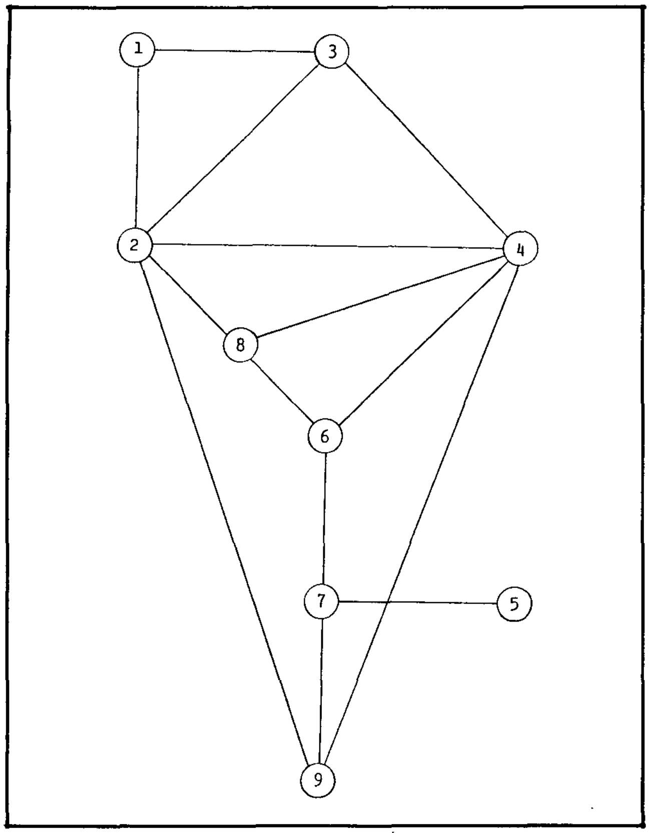

FIGURE 1. Graph for Example 1.

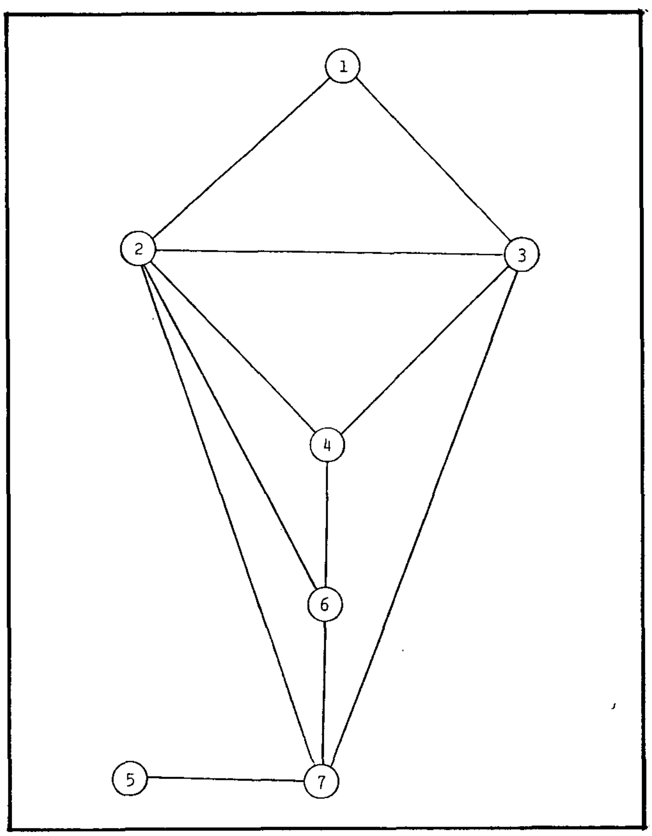

FIGURE 2. Graph for Example 2.

Example 2
Consider the graph shown in Figure 2. Let the coloration order be $x_1, x_2, \ldots, x_7$. Assume x_1, x_2, x_3 to be the initial clique and 5 to be an upper bound for the chromatic number. Without any backtrack, the Brelaz algorithm finds the four-coloration 1, 2, 3, 1, 1, 3, 4. As described in Step 7 of the algorithm we must now backtrack from x_7 to x_6. From there the next backtrack leads us to x_4 where the algorithm stops. The exact solution 1, 2, 3, 1, 2, 3, 1 has been eliminated by the algorithm.

A CORRECT VERSION OF BROWN'S MODIFIED ALGORITHM

Let n be the number of vertices of the graph. Let w be the dimension of an initial clique; let q be the number of colors used by a heuristic. Without loss of generality, assume $w < q$ and $x_1, \ldots, x_w$ to be the clique vertices.

BEGIN
Initialize back = false, $k = w + 1$. Label all the clique vertices.
DO FOREVER
  IF not back
  THEN determine u_k as the number of colors used for the actual partial solution of level $k - 1$; determine $U(x_k)$ as the set of colors from $\{1, \ldots, \min(u_k + 1, q - 1)\}$ which are not used in the actual partial solution of level $k - 1$ for a neighbor of x_k;
  ELSE let c be the actual color of x_k and write $U(x_k) = U(x_k) - \{c\}$; remove the label from x_k if there is one;
  FI;
  IF $U(x_k) \neq \emptyset$
  THEN determine i as the minimal color in $U(x_k)$; color vertex x_k with color i, i.e., i now is the actual color of x_k; set $k = k + 1$;
    IF $k > n$
    THEN you have found a new solution; let s be the number of colors used for this solution and set $q = s$;
EXIT IF $q = w$;
  determine k as the minimal rank among all q-colored vertices and remove all labels from the vertices $x_k, \ldots, x_n$;
    set back = true;
    ELSE set back = false;
    FI;
  ELSE set back = true;
  FI;
  IF back
  THEN call procedure label (x_k); determine k as the maximal rank among all labeled vertices;
EXIT IF $k \leqq w$;
  FI;
OD;
STOP;
PROCEDURE label (x_k):
  label all the unlabeled vertices which possess all the following properties
  (i) smaller rank than rank of x_k,
  (ii) adjacent to x_k,
  (iii) minimal rank among all the vertices of their color which are adjacent to x_k;
END;

THEOREM. *Brown's modified algorithm is an exact method for coloring the vertices of a graph.*

PROOF. First it is clear that in all situations where the variable back is set to the value true, a backtrack is necessary.

Therefore consider the situation where there is a backtrack from x_k to $x_{k'}$ with $k > k'$. We have to show that there is no s-coloration S with $s < q$ and $S \supset \{(x_1, c(1)), \ldots, (x_{k'}c(k'))\}$ where this set means the actual partial solution of level k'. This statement is true if $k = n$ and $x_{k'}$ is the vertex with minimal rank among all vertices of color q.

Therefore the only backtracks of interest are those in which $x_{k'}$ is the vertex of maximal rank among all labeled ones. Let m be the number of those backtracks and let $L_m = \{(x_i, c(x_i)) : x_i$ belongs to the set of labeled vertices when the mth backtrack is made$\}$.

We will now show by induction on m that there is no solution S with $s < q$ and $L_m \subset S$.

If $m = 1$, then all the colors $1, \ldots, q - 1$ are prohibited for x_k by a neighbor of x_k which is already colored with one of them. Therefore you cannot find any solution until one of these neighbors is given another color.

If $m > 1$, we have to consider two situations: (i) Analogously to the case when $m = 1$ all the colors $1, \ldots, q - 1$ are prohibited for x_k by a colored neighbor. (ii) There exists at least one color $c \leqq \min(u_k + 1, q - 1)$ which is not prohibited for x_k by a colored neighbor but which has been proved to be a flop and therefore been removed from $U(x_k)$ by an earlier backtrack $m' < m$; i.e., $x_k, c \in L_{m'}$. If there were a solution S with $s < q$ and $L_m \subset S$, then x_k would have such a color c; i.e., $(x_k, c) \in S$. This together with $(x_k, c) \in L_{m'}$ and $L_{m'} - \{(x_k, c)\} \subset L_m \subset S$ implies $L_{m'} \subset S$, which is a contradiction to our induction hypothesis. □

Remark. You can modify this algorithm if in situation (i) of the proof you go back to the vertex with minimal rank among all vertices labeled by x_k.

BROWN'S MODIFIED ALGORITHM WITH LOOK-AHEAD RULE

In this section, we want to show how we can build Brown's look-ahead rule (see [2]) into the modified algorithm.

Let $w < q$ and $x_1, \ldots, x_w$ be as above.

BEGIN
Initialize back = false, block = false, $k = w + 1$; label all the clique vertices;
DO FOREVER
  IF not back
  THEN determine u_k and $U(x_k)$; for each $c \in U(x_k)$ determine the number of uncolored neighbors of x_k for which c still is an unprohibited color (hereafter called number of preventions) and the number of uncolored neighbors of x_k where a coloration of x_k with c would block these neighbors, i.e., c is the only unprohibited color for these neighbors among $\{1, \ldots, q - 1\}$ (hereafter called number of blockings); order the colors $c \in U(x_k)$, first by the number of blockings, and second by the number of preventions;
  ELSE let c be the actual color of x_k and write $U(x_k) = U(x_k) - \{c\}$; remove the label from x_k if there is one;
  FI;
  IF $U(x_k) \neq \emptyset$
  THEN
    determine i as the color of minimal order in $U(x_k)$;
    IF i is not a blocking color
    THEN color vertex x_k with color i; set $k = k + 1$;
      IF $k > n$
      THEN you have found a new solution; set $q = s$;
EXIT IF $q = w$;
        determine k as the minimal rank among all q-colored vertices and remove all labels from the vertices $x_k, \ldots, x_n$; set back = true;
      ELSE set back = false;
      FI;
    ELSE set back = true; set block = true;
      FI,
  ELSE set back = true;
  FI;
  IF back
  THEN
    IF block
    THEN for each blocking color $c \in U(x_k)$ determine all vertices x_c which would be blocked by c and call procedure label (x_c) for each of them; set block = false;
    FI;
    call procedure label (x_k); determine k as the maximal rank among all labeled vertices;
EXIT IF $k \leqq w$;
  FI;
OD;
STOP;
END;

THEOREM. *Brown's modified algorithm with look-ahead rule is an exact method for coloring the vertices of a graph.*

PROOF. The proof works rather similarly to the above one. □

Remark. You can modify the last algorithm if you call procedure label only for those vertices x_c which have minimal rank among all vertices which would be blocked by c.

REFERENCES

1. Brelaz, D. New methods to color the vertices of a graph. *Comm. ACM 22*, 4 (April 1979), 251–256.
2. Brown, J. R. Chromatic scheduling and the chromatic number problem. *Manage. Sci. 19*, 4 (Dec. 1972), I, 456–463.

CR Categories and Subject Descriptors: G. 2.2 [**Discrete Mathematics**]: Graph Theory—*graph algorithms*; I. 2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods and Search—*backtracking, graph and tree search strategies*

General Terms: Algorithms

Additional Key Words and Phrases: graph coloring, chromatic number, exact method, branch and bound, computed backtracking, look-ahead rule

Received 1/82; revised 5/82; accepted 9/82

CORRIGENDUM. *Human Aspects of Computing.*

Steven L. Sauter et al., Job and health implications of VDT use: Initial Results of the Wisconsin-NIOSH Study. Comm. ACM 26, 4 (April 1983), 284-294.

Page 290: Column 1: In line 36, "...in all but one case..." should read "...in only one case..."

Page 291: Column 2: In line 4, "...the effect is most pronounced for individuals with monofocal reading glasses, as opposed to bifocals or contact lenses." should read "...the effect is most pronounced for individuals with monofocal glasses or contact lenses, as opposed to multifocal lenses."