On edge orienting methods for graph coloring

Bernard Gendron[1]   Alain Hertz[2]   Patrick St-Louis[1]

March 20, 2006

Abstract

We consider the problem of orienting the edges of a graph so that the length of a longest path in the resulting digraph is minimum. As shown by Gallai, Roy and Vitaver, this edge orienting problem is equivalent to finding the chromatic number of a graph. We study various properties of edge orienting methods in the context of local search for graph coloring. We then exploit these properties to derive four tabu search algorithms, each based on a different neighborhood. We compare these algorithms numerically to determine which are the most promising and to give potential research directions.

1 Introduction

All graphs in this paper have no loops and no multiple edges. For a graph G, we denote $V(G)$ its vertex set and $E(G)$ its edge set. A *partial* graph of G is a graph obtained from G by removing some edges. A k-coloring of G is a function $c : V(G) \rightarrow \{1, \ldots, k\}$. It is said *legal* if $c(i) \neq c(j)$ for all edges (i, j) in $E(G)$. The smallest integer k such that a legal k-coloring exists for G is the chromatic number $\chi(G)$ of G. Finding the chromatic number of a given graph is known as the *graph coloring problem*, and is NP-hard [5]. Although many exact algorithms have been devised for this particular problem [2, 7, 8, 9, 10], such algorithms can only be used to solve small instances.

A directed graph (or just *digraph*) is a graph with an orientation on each edge. An edge (u, v) oriented from u to v is called an *arc*, is denoted $u \rightarrow v$, and u is its *tail* while v is its *head*. An *orientation* of a graph G is a digraph, denoted $\vec{G}$, obtained from G by choosing an orientation $u \rightarrow v$ or $v \rightarrow u$ for each edge $(u, v) \in E(G)$. By removing some arcs in $\vec{G}$, one gets a *partial* digraph of $\vec{G}$. A *path* is a digraph, denoted $(v_0 \rightarrow \cdots \rightarrow v_{k-1})$, with vertices $v_0, \cdots, v_{k-1}$ and arcs $v_{i-1} \rightarrow v_i$ $(i = 1, \cdots, k-1)$, and its *length* is its number of vertices. We denote $\lambda(\vec{G})$ the length of a longest path in $\vec{G}$. Also, we denote $d_{\vec{G}}^+(v)$ the length of a longest path in $\vec{G}$ starting at v, while $d_{\vec{G}}^-(v)$ denotes the length of a longest path ending at v. Hence, $\lambda(\vec{G}) = d_{\vec{G}}^-(u) + d_{\vec{G}}^+(v)$ for every arc $u \rightarrow v$ lying on a longest path in $\vec{G}$.

---

[1]Département d'informatique et de recherche opérationnelle, Université de Montréal, Canada, gendron,stlouip@iro.umontreal.ca

[2]Département de Mathématiques et de Génie Industriel, École Polytechnique, Montréal, Canada, Alain.Hertz@gerad.ca

A *source* is a vertex v with $d_{\vec{G}}^-(v) = 0$ while a *sink* is a vertex v with $d_{\vec{G}}^+(v) = 0$. A *circuit* is a digraph, denoted $(v_0 \to \cdots \to v_{k-1} \to v_0)$, and obtained by adding an arc $v_{k-1} \to v_0$ to a path $(v_0 \to \cdots \to v_{k-1})$. A *dicycle* is an orientation of a cycle, which means that every circuit is a dicycle. If a dicycle $\vec{C}$ is not a circuit, then it contains a set $A = \{v_0, \cdots, v_{k-1}\}$ of sources and a set $B = \{w_0, \cdots, w_{k-1}\}$ of sinks ($k > 0$) such that there is a path linking v_i to w_i and one linking v_i to w_{i-1} for every $i = 0, \cdots k-1$ (the indices being taken modulo k), and $\vec{C}$ is the union of these paths. We use the notation $\vec{C} = (v_0 \Rightarrow w_0 \Leftarrow \cdots v_{k-1} \Rightarrow w_{k-1} \Leftarrow v_0)$. This is illustrated on Figure 1 where the set of sources on $\vec{C}$ is $A = \{g, d\}$ and the set of sinks is $B = \{b, f\}$. Notice that the notation is not unique since $(g \Rightarrow b \Leftarrow d \Rightarrow f \Leftarrow g) = (d \Rightarrow f \Leftarrow g \Rightarrow b \Leftarrow d) = (g \Rightarrow f \Leftarrow d \Rightarrow b \Leftarrow g) = (d \Rightarrow b \Leftarrow g \Rightarrow f \Leftarrow d)$ in the example of Figure 1. Notice also that if $\vec{C}$ is a dicycle in a digraph $\vec{G}$, then a source (sink) in $\vec{C}$ is not necessarily a source in $\vec{G}$, as illustrated by vertex d in Figure 1.

A graph G with
a cycle C=(b,c,d,f,g,h,b)

A circuit-free orientation G of $\vec{G}$
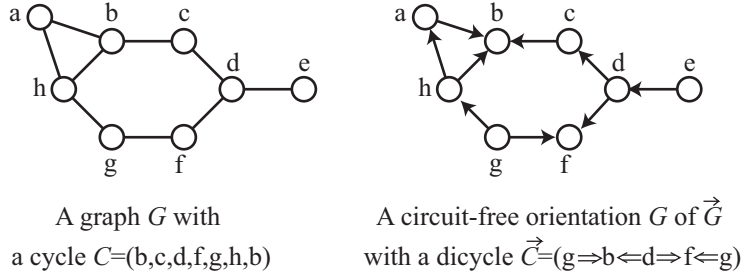with a dicycle $\vec{C}$=(g⇒b⇐d⇒f⇐g)

Figure 1. Illustration of a notation

Finally, we denote $\Omega(G)$ the set of circuit-free orientations of G and, for a fixed integer k, we denote $\Pi_k(G)$ the set of circuit-free orientations $\vec{H}$ of a partial graph H of G such that $\lambda(\vec{H}) \leq k$.

Gallai [4], Roy [11] and Vitaver [13] have independently proven the following famous theorem.

Theorem 1 *[4, 11, 13] The length of a longest path in an orientation of a graph G is at least equal to the chromatic number of G.*

It follows from Theorem 1 that the problem of determining the chromatic number of a graph G is equivalent to the problem of orienting the edges of G so that the resulting digraph $\vec{G}$ is circuit-free and $\lambda(\vec{G})$ is minimum. Indeed, given a $\chi(G)$-coloring c of a graph G, one can easily construct a circuit-free orientation $\vec{G}$ with $\lambda(\vec{G}) = \chi(G)$ by simply orienting each edge (u, v) from u to v if and only if $c(u) < c(v)$. Conversely, given a circuit-free orientation $\vec{G}$ of G, one can build a $\lambda(\vec{G})$-coloring of G by assigning to each vertex v a color $c(v)$ equal to the length $d_{\vec{G}}^-(v)$ of a longest path ending at v in $\vec{G}$. Hence, one can state the following corollary of Theorem 1.

Corollary 2 $\qquad \chi(G) = \min_{\vec{G} \in \Omega(G)} \lambda(\vec{G})$

It is interesting to note that the problem of determining a graph in $\Omega(G)$ with maximum $\lambda(\vec{G})$ is also NP-hard since any algorithm that would solve this problem would also solve the

Hamiltonian path problem which is known to be NP-hard [5]. Indeed, a graph G contains a Hamiltonian path if and only if its edges can be oriented so that the resulting digraph $\vec{G}$ is circuit-free and $\lambda(\vec{G}) = \mid V(G) \mid$.

Since exact methods can only solve small graph coloring instances, heuristic methods are needed to get an upper bound on the chromatic number. As shown in a recent survey [3], most recent efficient graph coloring heuristics are either local search methods (e.g., tabu search, simulated annealing) or hybrid algorithms that combine a local search with a population-based method (e.g., genetic algorithm, adaptive memory programming). Barbosa et al. [1] have implemented a genetic algorithm that aims to minimize $\lambda(\vec{G})$. They consider a population of graph orientations chosen in $\Omega(G)$. Each member of the population is coded as a partial order of the vertices, where $u \to v$ means that u preceeds v. A crossover operator combines two partial orders of the vertices for creating new members in the population (i.e, new graph orientations $\vec{G} \in \Omega(G)$). To our knowledge, no other evolutionary graph coloring algorithm based on edge orienting methods has ever been proposed, and we are not aware of any local search method that uses graph orientations to determine the chromatic number of a graph. The aim of this paper is to help bridging this gap by providing an analysis of the main ingredients that may be useful in the design of a local search graph coloring algorithm based on edge orienting methods.

Local search methods can be described as follows. Let S be the set of solutions to a combinatorial optimization problem. For a solution $s \in S$, let $N(s)$ denote the *neighborhood* of s which is defined as the set of *neighbor solutions* in S obtained from s by performing a *local change* on it. Local search techniques visit a sequence $s_0, \ldots, s_n$ of solutions, where s_0 is an initial solution and $s_{i+1} \in N(s_i)$ $(i = 1, \ldots, n-1)$. When designing a local search algorithm for solving a particular problem, one has to define the search space to be explored, the evaluation function to be minimized, and the neighborhood function. This triplet is called a *search strategy*. In their recent survey, Galinier and Hertz [3] propose to classify the search strategies that have proven to be efficient for the graph coloring problem into four categories :

- The *legal strategy*: The search space S contains all legal colorings and the goal is to find a solution $c \in S$ that uses as few colors as possible.

- The *k-fixed partial legal strategy*: The number k of colors is fixed, the search space S contains all partial legal k-colorings, and the goal is to determine a solution $c \in S$ in which all vertices are colored.

- The *k-fixed penalty strategy*: The number k of colors is fixed, the search space S contains all (not necessarily legal) k-colorings, and the goal is to determine a legal k-coloring $c \in S$.

- The *penalty strategy*: The search space S contains all (not necessarily legal) colorings and the goal is to determine a legal coloring $c \in S$ that uses as few colors as possible.

There are at least two additional categories of local search strategies based on edge orienting methods that have never been tested and could prove successful for graph coloring.

- The *total orienting strategy*: The search space S contains all circuit-free graph orientations $\vec{G}$ of G (i.e. $S = \Omega(G)$), and the goal is to minimize $\lambda(\vec{G})$.

- The *k-fixed partial orienting strategy*: An upper bound k on $\lambda(\vec{G})$ is fixed, the search space S contains all circuit-free graph orientations $\vec{H}$ of a partial graph H of G with $\lambda(\vec{H}) \leq k$ (i.e., $S = \Pi_k(G)$), and the aim is to determine a solution $\vec{H}$ having a maximum number of arcs.

For the *total orienting strategy*, a neighbor of a solution $\vec{H} \in \Omega(G)$ can be obtained by modifying the orientation of a subset of arcs, without creating circuits. For the *k-fixed partial orienting strategy*, a neighbor of a solution $\vec{H} \in \Pi_k(G)$ can be obtained by adding and orienting some edges that belong to G but not to H, and by then removing some arcs (if necessary) so that the resulting digraph also belongs to $\Pi_k(G)$. In summary, neighbor solutions in the above orienting strategies are obtained by changing the orientation of some arcs, or by adding some arcs and removing some others. We study in the next section some properties related to such changes. In particular, we characterize those that produce a circuit-free neighbor $\vec{H}$, and we compare $\lambda(\vec{G})$ with $\lambda(\vec{H})$. This theoretical study will help us to design and implement four basic tabu search algorithms for graph coloring based on orienting strategies. The four algorithms are described in Section 3 and compared numerically in Section 4 in order to determine which ones are the most promising. The algorithms we present illustrate the kind of methods one can design based on the theoretical results presented in Section 2. The development of more elaborate approaches is a topic for future study.

2 Properties

We start this section by giving some properties related to the modification of the orientation of some arcs in a digraph.

Property 3 *Let $\vec{G}$ be a circuit-free orientation of a graph G and let $\vec{H}$ be the digraph obtained from $\vec{G}$ by changing the orientation of a subset F of arcs. Then $\vec{H}$ contains a circuit if and only if there is a dicycle $\vec{C} = (v_0 \Rightarrow w_0 \Leftarrow \cdots v_{k-1} \Rightarrow w_{k-1} \Leftarrow v_0)$ in $\vec{G}$ such that no arc on the paths from v_i to w_i belongs to F and every arc on the paths from v_i to w_{i-1} belongs to F $(i = 0, \cdots k-1)$.*

Proof.

Assume there is a dicycle $\vec{C} = (v_0 \Rightarrow w_0 \Leftarrow \cdots v_{k-1} \Rightarrow w_{k-1} \Leftarrow v_0)$ in $\vec{G}$ such that no arc on the paths from v_i to w_i belongs to F and every arc on the paths from v_i to w_{i-1} belongs to F. Then $\vec{C}$ is transformed into a circuit in $\vec{H}$.

Suppose now that $\vec{H}$ contains a circuit $(x_0 \rightarrow \cdots x_{r-1} \rightarrow x_0)$. Let I be the set containing all vertices x_i on the circuit such that $x_i \rightarrow x_{i-1} \in F$ and $x_{i+1} \rightarrow x_i \notin F$. Similarly, define J as the set containing all x_j with $x_j \rightarrow x_{j-1} \notin F$ and $x_{j+1} \rightarrow x_j \in F$. Then the circuit in $\vec{H}$ corresponds to a dicycle $\vec{C}$ in $\vec{G}$ with I as set of sources and J as set of sinks. Hence, $\vec{C} = (v_0 \Rightarrow w_0 \Leftarrow \cdots v_{k-1} \Rightarrow w_{k-1} \Leftarrow v_0)$ with $I = \{v_0, \cdots, v_{k-1}\}$ and $J = \{w_0, \cdots, w_{k-1}\}$. ∎

The next two corollaries are about the special case where $|F| = 1$.

Corollary 4 *Let $\vec{G}$ be a circuit-free orientation of a graph G and let $\vec{H}$ be the digraph obtained from $\vec{G}$ by changing the orientation of exactly one arc from $u \to v$ to $v \to u$. Then $\vec{H}$ contains a circuit if and only there is a path of length at least 3 linking u to v in $\vec{G}$.*

Proof. According to Property 3, $\vec{H}$ contains a circuit if and only if there is a dicycle $(v_0 \Rightarrow w_0 \Leftarrow v_0)$ in $\vec{G}$. Since G has no multiple edges, the path from v_0 to w_0 in $\vec{G}$ has length at least 3. ∎

Corollary 5 *If $\vec{H}$ if obtained from a circuit-free orientation $\vec{G}$ of G by changing the orientation of exactly one arc $u \to v$ with $d_{\vec{G}}^-(v) = d_{\vec{G}}^-(u) + 1$, then $\vec{H}$ is circuit-free.*

Proof. From the previous Corollary, it is sufficient to observe that there is no path of length > 2 linking u to v in $\vec{G}$, else $d_{\vec{G}}^-(v) \geq d_{\vec{G}}^-(u) + 2$. ∎

The next properties deal with the special case where F contains arcs lying on longest paths. The first part of Property 6 (i.e., $\vec{H}$ is circuit-free) was also proved in [12].

Property 6 *If $\vec{H}$ if obtained from a circuit-free orientation $\vec{G}$ of G by changing the orientation of exactly one arc $u \to v$ on a longest path, then*

- *$\vec{H}$ is circuit-free*

- *$\lambda(\vec{H}) \leq \lambda(\vec{G}) + 2$.*

- *if $\lambda(\vec{H}) > \lambda(\vec{G})$ then at least one the following situations must occur:*
  - *there is a second arc $w \to v$ entering v that also belongs to a longest path in $\vec{G}$;*
  - *there is a second arc $u \to w$ leaving u that also belongs to a longest path in $\vec{G}$.*

Proof. Since $u \to v$ is on a longest path in $\vec{G}$, there is no path of length > 2 linking u to v, and we therefore know from Corollary 4 that $\vec{H}$ is circuit-free.

Since $d_{\vec{G}}^+(u) = d_{\vec{G}}^+(v) + 1, d_{\vec{G}}^-(v) = d_{\vec{G}}^-(u) + 1, d_{\vec{H}}^-(v) \leq d_{\vec{G}}^-(v)$, and $d_{\vec{H}}^+(u) \leq d_{\vec{G}}^+(u)$, we know that the length of a longest path in $\vec{H}$ that contains $v \to u$ has length $d_{\vec{H}}^-(v) + d_{\vec{H}}^+(u) \leq d_{\vec{G}}^-(u) + d_{\vec{G}}^+(v) + 2 = \lambda(\vec{G}) + 2$. Since all paths in $\vec{H}$ that do not contain $v \to u$ have a length $\leq \lambda(\vec{G})$, we conclude that $\lambda(\vec{H}) \leq \lambda(\vec{G}) + 2$.

If $\lambda(\vec{H}) > \lambda(\vec{G})$, then every longest path in $\vec{H}$ necessarily contains the arc $v \to u$, hence $\lambda(\vec{H}) = d_{\vec{H}}^-(v) + d_{\vec{H}}^+(u)$. If $u \to v$ is the unique arc entering v and the unique arc leaving u that belongs to a longest path in $\vec{G}$, then $d_{\vec{H}}^-(v) \leq d_{\vec{G}}^-(v) - 1$ and $d_{\vec{H}}^+(u) \leq d_{\vec{G}}^+(u) - 1$, which means that $\lambda(\vec{H}) \leq d_{\vec{G}}^-(v) + d_{\vec{G}}^+(u) - 2 = \lambda(\vec{G})$, a contradiction. ∎

When designing a local search algorithm, it is important to ensure that given any initial solution s_0, there is a sequence of neighbor solutions leading to an optimal one (i.e., a sequence $s_0, \cdots, s_n$ with $s_i \in N(s_{i-1})$ and where s_n is optimal). The next property demonstrates that this requirement is satisfied for the *total orienting strategy* when a neighbor is obtained by modifying the orientation of one arc on a longest path. For a digraph $\vec{H} \in \Omega(G)$, let $N_1(\vec{H})$ denote the set of digraphs that can be obtained from $\vec{G}$ by modifying the orientation of exactly one arc on a longest path in $\vec{H}$. We know from Property 6 that every digraph in $N_1(\vec{H})$ belongs to $\Omega(G)$.

Property 7 *For every digraph $\vec{G} \in \Omega(G)$ there is a sequence $(\vec{H}_0, \cdots, \vec{H}_n)$ of digraphs such that $\vec{H}_0 = \vec{G}$, $\lambda(\vec{H}_n) = \chi(G)$, and each $\vec{H}_i$ belongs to $N_1(\vec{H}_{i-1})$.*

Proof. Let $\vec{G}^*$ be an orientation of G with $\lambda(\vec{G}^*) = \chi(G)$. Also, for an orientation $\vec{H}_i \in \Omega(G)$, let $A(\vec{H}_i)$ denote the set of arcs in $\vec{H}_i$ that have the opposite orientation in $\vec{G}^*$. If $\lambda(\vec{H}_i) > \chi(G)$, then every longest path in $\vec{H}_i$ contains at least one arc in $A(\vec{H}_i)$. Let $\vec{H}_{i+1}$ be the digraph obtained from $\vec{H}_i$ by changing the orientation of one such arc. One can repeat this process until one reaches a digraph $\vec{H}_n$ with $\lambda(\vec{H}_n) = \chi(G)$. This will occur in at most $\mid A(\vec{H}_0) \mid$ steps since $\mid A(\vec{H}_{i+1}) \mid = \mid A(\vec{H}_i) \mid -1$ and $\vec{H}_n = \vec{G}^*$ when $\mid A(\vec{H}_n) \mid = 0$. ∎

Notice that for reaching an optimal solution $\vec{H}_n$ starting from a solution $\vec{H}_0$, it can be necessary to visit a solution $\vec{H}_i$ with $\lambda(\vec{H}_i) > \lambda(\vec{H}_0)$. This is illustrated in Figure 2. The digraph $\vec{H}_0$ has two longest paths $(e \to d \to b \to a)$ and $(e \to c \to b \to a)$ of length 4. One can easily verify that by changing the orientation of $e \to d$, $e \to c$, $d \to b$, or $c \to b$, one gets a digraph $\vec{H}$ with a longest path $(c \to e \to d \to b \to a)$ of length $\lambda(\vec{H}) = 5$. The unique arc on a longest path in $\vec{H}_0$ that can be reversed without increasing the length of a longest path is the arc $b \to a$. But the digraph obtained by reversing this arc is similar to $\vec{H}_0$, where a and b have exchanged their role. Hence, in order to reach an optimal orientation $\vec{H}_n$ of G with $\lambda(\vec{H}_n) = 3$ (as the rightmost digraph of Figure 2 that contains two longest paths $(c \to b \to a)$ and $(d \to b \to a)$), one has to visit a digraph with longest path of length $5 > 4 = \lambda(\vec{H}_0)$.
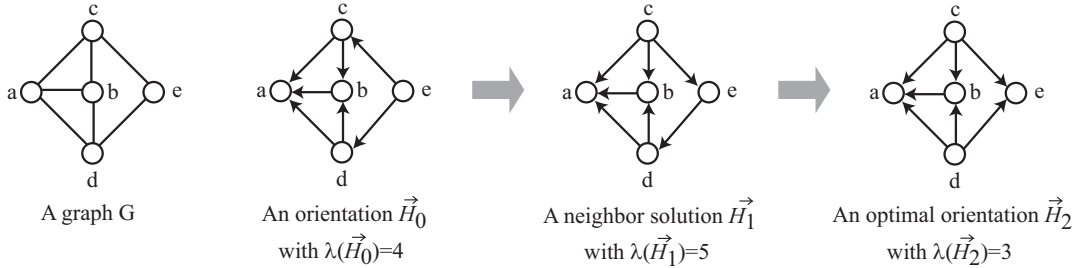

Figure 2. A sequence of neighbor solutions

For a digraph $\vec{G}$, let $W(\vec{G})$ denote its set of vertices on a longest path. Also, for a vertex $v \in W(\vec{G})$, let $A_{\vec{G}}^-(v)$ and $A_{\vec{G}}^+(v)$ denote the set of arcs on longest paths in $\vec{G}$ having v as head and as tail, respectively.

Property 8 *Let $\vec{G}$ be a circuit-free orientation of G, and let v be a vertex in $W(\vec{G})$. If $\vec{H}$ is obtained from $\vec{G}$ by changing the orientation of a subset of arcs in $A_{\vec{G}}^-(v)$ or a subset of arcs in $A_{\vec{G}}^+(v)$, then $\vec{H}$ is also circuit-free.*

Proof. Let F be the set of arcs in $\vec{G}$ that have been modified in $\vec{G}$ to get $\vec{H}$. If $\vec{H}$ contains a circuit, then we know from Property 3 that there is a dicycle $\vec{C} = (v_0 \Rightarrow w_0 \Leftarrow \cdots v_{k-1} \Rightarrow w_{k-1} \Leftarrow v_0)$ in $\vec{G}$ such that no arc on the paths from v_i to w_i belongs to F and every arc on the paths from v_i to w_{i-1} belongs to F. Since all arcs in F have the same tail (or the same head), such a dicycle $\vec{C}$ contains only one arc of F, which means that $\vec{C}$ is the union of the arc $v_0 \to w_0$ in F with a directed path $\vec{P}$ linking v_0 to w_0 having no arc in F. Since G has no multiple edges, $\vec{P}$ has length at least three, which contradicts the fact that $v_0 \to w_0$ is on a longest path in $\vec{G}$. ∎

6

Property 9 *Let $\vec{G}$ be a circuit-free orientation of G, and let v be a vertex in $W(\vec{G})$. If $\vec{H}$ is obtained from $\vec{G}$ by changing the orientation of all arcs in $A_{\vec{G}}^-(v)$ or all arcs in $A_{\vec{G}}^+(v)$, then*

- *$\vec{H}$ is circuit-free;*
- *$\lambda(\vec{H}) \leq \lambda(\vec{G}) + 1$.*

Proof. We know from Property 8 that $\vec{H}$ is circuit-free. Let $\vec{P}$ be a longest path in $\vec{H}$. If all arcs on $\vec{P}$ have the same orientation as in $\vec{G}$, then $\lambda(\vec{H}) \leq \lambda(\vec{G})$. Otherwise, let F be the set of arcs that have been modified in $\vec{G}$ to get $\vec{H}$. Since all arcs in F have the same tail or the same head, $\vec{P}$ contains exactly one arc $x \to y$ that was originally oriented from y to x in $\vec{G}$. Hence, $v = x$ and $F = A_{\vec{G}}^-(v)$, or $v = y$ and $F = A_{\vec{G}}^+(v)$. We therefore have either $d_{\vec{H}}^-(x) < d_{\vec{G}}^-(x)$ and $d_{\vec{H}}^+(y) \leq d_{\vec{G}}^+(y)$, or $d_{\vec{H}}^+(y) < d_{\vec{G}}^+(y)$ and $d_{\vec{H}}^-(x) \leq d_{\vec{G}}^-(x)$. So $\lambda(\vec{H}) = d_{\vec{H}}^-(x) + d_{\vec{H}}^+(y) < d_{\vec{G}}^-(x) + d_{\vec{G}}^+(y) = \lambda(\vec{G}) + 2$. ∎

We now prove a result similar to Property 7. For a digraph $\vec{H} \in \Omega(G)$, let $N_2(\vec{H})$ denote the set of digraphs that can be obtained from $\vec{G}$ by changing the orientation of all arcs in $A_{\vec{G}}^-(v)$ or all arcs in $A_{\vec{G}}^+(v)$, for some $v \in W(\vec{G})$. We know from Property 9 that every digraph in $N_2(\vec{H})$ belongs to $\Omega(G)$.

Property 10 *For every $\vec{G} \in \Omega(G)$ there is a sequence $(\vec{H}_0, \cdots, \vec{H}_n)$ of digraphs such that $\vec{H}_0 = \vec{G}$, $\lambda(\vec{H}_n) = \chi(G)$, and each $\vec{H}_i$ belongs to $N_2(\vec{H}_{i-1})$.*

Proof. Let $\vec{G}^*$ be an orientation of G with $\lambda(\vec{G}^*) = \chi(G)$. Also, for an orientation $\vec{H}_i \in \Omega(G)$, let $A(\vec{H}_i)$ denote the set of arcs in $\vec{H}_i$ that have the opposite orientation in $\vec{G}^*$. If $\lambda(\vec{H}_i) > \chi(G)$, then there is a vertex $v \in W(\vec{H}_i))$ such that all arcs in $A_{\vec{H}_i}^+(v)$ have the opposite orientation in $\vec{G}^*$. Indeed, assume by contradiction that each vertex $v \in W(\vec{H}_i))$ with $A_{\vec{H}_i}^+(v) \neq \emptyset$ is the tail of at least one arc $v \to w$ in $\vec{H}_i$ that has the same orientation as in $\vec{G}^*$. Consider any vertex v_1 in $\vec{H}_i$ with $d_{\vec{H}_i}^+(v_1) = \lambda(\vec{H}_i)$. Then each vertex v_j ($j = 1, \cdots, \lambda(\vec{H}_i) - 1$) is the tail of an arc $v_j \to v_{j+1}$ in $A_{\vec{H}_i}^+(v_j)$ that has the same orientation as in $\vec{G}^*$. Hence $(v_1, \cdots, v_{\lambda(\vec{H}_i)})$ is a path of length $\lambda(\vec{H}_i) > \lambda(\vec{G}^*)$ in $\vec{G}^*$, a contradiction.

So let v be a vertex in $W(\vec{H}_i))$ such that all arcs in $A_{\vec{H}_i}^+(v)$ have the opposite orientation in $\vec{G}^*$, and let $\vec{H}_{i+1}$ be the digraph obtained from $\vec{H}_i$ by changing the orientation of all arcs in $A_{\vec{H}_i}^+(v)$. One can repeat this process until one reaches a digraph $\vec{H}_n$ with $\lambda(\vec{H}_n) = \chi(G)$. This will occur in at most $\mid A(\vec{H}_0) \mid$ steps since $\mid A(\vec{H}_{i+1}) \mid < \mid A(\vec{H}_i) \mid$ and $\vec{H}_n = \vec{G}^*$ when $\mid A(\vec{H}_n) \mid = 0$. ∎

Notice that we could have proven the above Property by showing, in a very similar way, that if $\lambda(\vec{H}_i) > \chi(G)$, then there is a vertex $v \in W(\vec{H}_i))$ such that all arcs in $A_{\vec{H}_i}^-(v)$ have the opposite orientation in $\vec{G}^*$.

For a digraph $\vec{G}$ and an integer $\ell \in \{1, \cdots, \lambda(\vec{G}) - 1\}$, let $\vec{L}_\ell(\vec{G})$ denote the partial digraph of $\vec{G}$ containing only those arcs $u \to v$ on longest paths such that $d_{\vec{G}}^-(u) = \ell$ and $d_{\vec{G}}^-(v) = \ell + 1$.

Property 11 *Let $\vec{G}$ be a circuit-free orientation of a graph G, and consider any integer $\ell \in \{1, \cdots, \lambda(\vec{G}) - 1\}$. If $\vec{H}$ if obtained from $\vec{G}$ by changing the orientation of all arcs in a subset of connected components of $\vec{L}_\ell(\vec{G})$, then*

- $\vec{H}$ *is circuit-free;*

- $\lambda(\vec{H}) \le \lambda(\vec{G})$.

Proof. Let F denote the set of arcs that have been modified in $\vec{G}$ to obtain $\vec{H}$. If $\vec{H}$ contains a circuit, then we know from Property 3 that there is a dicycle $\vec{C} = (v_0 \Rightarrow w_0 \Leftarrow \cdots v_{k-1} \Rightarrow w_{k-1} \Leftarrow v_0)$ in $\vec{G}$ such that no arc on the paths from v_i to w_i belongs to F and every arc on the paths from v_i to w_{i-1} belongs to F. Hence, $d_{\vec{G}}^-(v_i) = \ell$ and $d_{\vec{G}}^-(w_i) = \ell + 1$ for $i = 0, \cdots, k-1$, which means that each path from v_i to w_i and from v_i to w_{i-1} contains a single arc. Since G has no multiple edges, we have $k > 0$. So there are two longest paths $(x_1 \to \cdots \to x_\ell = v_0 \to x_{\ell+1} = w_{k-1} \to \cdots \to x_{\lambda(\vec{G})})$ and $(y_1 \to \cdots \to y_\ell = v_1 \to y_{\ell+1} = w_0 \to \cdots \to y_{\lambda(\vec{G})})$ in $\vec{G}$ containing $v_0 \to w_{k-1}$ and $v_1 \to w_0$, respectively. But this means that $(x_1 \to \cdots \to x_\ell = v_0 \to y_{\ell+1} = w_0 \to \cdots \to y_{\lambda(\vec{G})})$ is a longest path in $\vec{G}$ containing $v_0 \to w_0$, which implies that $v_0 \to w_0$ is in the same connected component of $\vec{L}_\ell(\vec{G})$ as $v_0 \to w_{k-1}$ and $v_1 \to w_0$. Hence, $v_0 \to w_0 \in F$, a contradiction. So $\vec{H}$ is circuit-free.

Assume now $\lambda(\vec{H}) > \lambda(\vec{G})$ and let $\vec{P} = (x_1 \to \cdots \to x_k)$ be a path in $\vec{H}$ of length $k > \lambda(\vec{G})$. Let A denote the set of indices i such that $x_{i+1} \to x_i \in F$. We have $\mid A \mid \ge 1$, else $\vec{P}$ also exists in $\vec{G}$. So let i be the smallest element in A. This means that $d_{\vec{G}}^-(x_i) = \ell + 1$, $d_{\vec{G}}^-(x_{i+1}) = \ell$, and there is a longest path $(y_1 \to \cdots \to y_\ell = x_{i+1} \to y_{\ell+1} = x_i \to \cdots \to y_{\lambda(\vec{G})})$ in $\vec{G}$ containing the arc $x_{i+1} \to x_i$. Suppose $\mid A \mid > 1$ and let j be the smallest element in A larger than i. Then there is a longest path $(z_1 \to \cdots \to z_\ell = x_{j+1} \to z_{\ell+1} = x_j \to \cdots \to z_{\lambda(\vec{G})})$ in $\vec{G}$ containing the arc $x_{j+1} \to x_j$. Since all arcs $x_r \to x_{r+1}$ of $\vec{P}$ with $i < r < j$ are also in $\vec{G}$, while $d_{\vec{G}}^-(x_{i+1}) = \ell$ and $d_{\vec{G}}^-(x_j) = \ell + 1$, we necessarily have $j = i + 2$. Hence, $(y_1 \to \cdots \to y_\ell = x_{i+1} \to z_{\ell+1} = x_{i+2} \to \cdots \to z_{\lambda(\vec{G})})$ is a longest path in $\vec{G}$ containing the arc $x_{i+1} \to x_{i+2}$. But this means that $x_{i+1} \to x_{i+2}$ is in the same connected component of $\vec{L}_\ell(\vec{G})$ as $x_{i+1} \to x_i$ and $x_{j+1} \to x_j$. Hence, $x_{i+1} \to x_{i+2} \in F$, a contradiction.

So we know that $A = \{i\}$ (i.e., A contains exactly one element). Since the path $(x_1 \to \cdots \to x_i = y_{\ell+1} \to \cdots \to y_{\lambda(\vec{G})})$ in $\vec{G}$ is of length $i + \lambda(\vec{G}) - (\ell + 1)$, we necessarily have $i \le \ell + 1$. If $i = \ell + 1$ then $x_{i-1} \to x_i$ is in the same connected component of $\vec{L}_\ell(\vec{G})$ as $x_{i+1} \to x_i$, which means that $x_{i-1} \to x_i \in F$, a contradiction. We therefore have $i \le \ell$. Moreover, since $(y_1 \to \cdots \to y_\ell = x_{i+1} \to \cdots \to x_k)$ is a path of length $\ell + k - (i+1)$ in $\vec{G}$, we have $k - i \le \lambda(\vec{G}) - \ell + 1$. Also, if $k - i = \lambda(\vec{G}) - \ell + 1$ then $x_{i+1} \to x_{i+2}$ is in the same connected component of $\vec{L}_\ell(\vec{G})$ as $x_{i+1} \to x_i$, which means that $x_{i+1} \to x_{i+2} \in F$, a contradiction. So finally, $k - i \le \lambda(\vec{G}) - \ell$, which implies that $\vec{P}$ is a path of length $k = i + (k - i) \le \ell + \lambda(\vec{G}) - \ell = \lambda(\vec{G})$, a contradiction. ∎

Let $N_3(\vec{H})$ denote the set of digraphs that can be obtained from $\vec{H}$ by choosing an integer $\ell \in \{1, \cdots, \lambda(\vec{H}) - 1\}$, and by changing the orientation of all arcs in a subset of connected components of $\vec{L}_\ell(\vec{H})$. One could think about proving a result similar to Properties 7 and 10, but with $\vec{H}_i \in N_3(\vec{H}_{i-1})$. However, the example in Figure 3 demonstrates that

there are digraphs $\vec{H} \in \Omega(G)$ such that there are no sequences $(\vec{H}_0, \cdots, \vec{H}_n)$ with $\vec{H}_0 = \vec{H}$, $\vec{H}_i \in N_3(\vec{H}_{i-1})(i = 1, \cdots n)$, and $\lambda(\vec{H}_n) = \chi(G)$. The left graph corresponds to a digraph $\vec{H}$ with $\lambda(\vec{H}) = 4$. In order not to overload the drawing, we indicate the value $d_{\vec{H}}^-(v)$ of each vertex v instead of representing each arc as a directed line. In other words, each arc should be directed from the smallest label to the largest one. As shown in Figure 3, each digraph $\vec{L}_\ell(\vec{H})$ ($\ell = 1, 2, 3$) contains exactly one connected component, and it is easy to verify that all digraphs in $N_3(\vec{G})$ are equivalent to $\vec{H}$, where the roles of the vertices with label $d_{\vec{H}}^-(v) = \ell$ are permuted with those with label $d_{\vec{H}}^-(v) = \ell + 1$. In summary, all digraphs in a sequence $(\vec{H}_0, \vec{H}_1, \cdots)$ with $\vec{H}_0 = \vec{H}$ and $\vec{H}_i \in N_3(\vec{H}_{i-1})$ are equivalent to $\vec{H}$ and have a longest path of length $\lambda(H_i) = 4$. However, an optimal orientation $\vec{G}$ of G with $\lambda(\vec{G}) = 3$ is represented on the right of Figure 3.
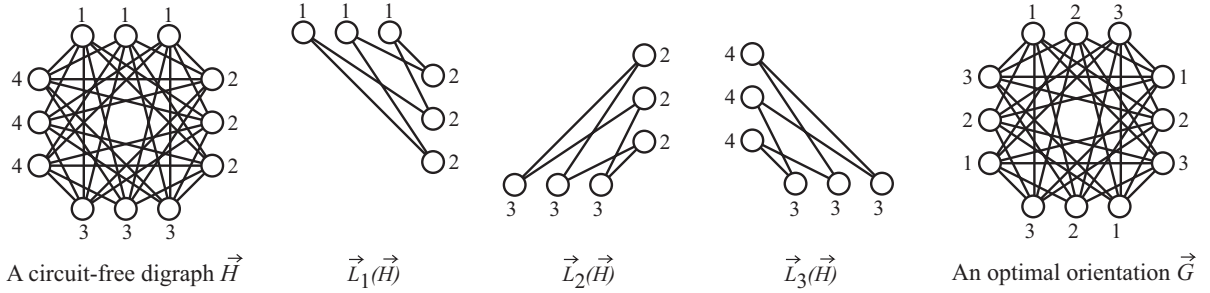

Figure 3. Illustration of the *total orienting strategy* with neighborhood N_3

We now give properties related to the addition or the removal of arcs.

Property 12 *Let $\vec{G}$ be a circuit-free orientation of a graph G, and let u and v be two non-adjacent vertices in G. If $d_{\vec{G}}^-(u) \leq d_{\vec{G}}^-(v) + 1$ then the graph $\vec{H}$ obtained by adding an arc $u \to v$ in $\vec{G}$ is circuit-free.*

Proof. If $\vec{H}$ contains a circuit $\vec{C} = (x_0 \to \cdots \to x_k \to x_0)$, then $\vec{C}$ contains the arc $u \to v$. Without loss of generality, we may assume $x_k = u$ and $x_0 = v$. But this means that $(x_0 \to \cdots \to x_k)$ is a path in $\vec{G}$. Since u is not adjacent to v in G, we have $k > 1$ which means that $d_{\vec{G}}^-(u) > d_{\vec{G}}^-(v) + 1$. ∎

By adding an arc between two non-adjacent vertices in a digraph $\vec{G}$, one gets a digraph $\vec{H}$. If $\vec{H}$ is circuit-free, then $\lambda(\vec{H}) \geq \lambda(\vec{G})$. The next property indicates when $\lambda(\vec{H})$ is strictly larger than $\lambda(\vec{G})$.

Property 13 *Let $\vec{G}$ be a circuit-free orientation of a graph G, and let $\vec{H}$ be a circuit-free digraph obtained from $\vec{G}$ by adding an arc $u \to v$ between two non-adjacent vertices u and v of G. Then $\lambda(\vec{H}) > \lambda(\vec{G})$ if and only if $d_{\vec{G}}^-(u) + d_{\vec{G}}^+(v) > \lambda(\vec{G})$.*

Proof. $\lambda(\vec{H}) > \lambda(\vec{G})$ if and only if there is a longest path in $\vec{H}$ containing $u \to v$ and having a length strictly larger than $\lambda(\vec{G})$. It is now sufficient to observe that the length of such a path is equal to $d_{\vec{G}}^-(u) + d_{\vec{G}}^+(v)$. ∎

Corollary 14 *Let $\vec{G}$ be a circuit-free orientation of a graph G, and let $\vec{H}$ be a circuit-free digraph obtained from $\vec{G}$ by adding an arc $u \to v$ between two non-adjacent vertices u and v of G with $d_{\vec{G}}^-(u) < d_{\vec{G}}^-(v)$. Then $\lambda(\vec{H}) = \lambda(\vec{G})$.*

Proof. Since $\lambda(\vec{G}) \geq d_{\vec{G}}^-(v) + d_{\vec{G}}^+(v) - 1$, we have $d_{\vec{G}}^-(u) + d_{\vec{G}}^+(v) \leq d_{\vec{G}}^-(u) - d_{\vec{G}}^-(v) + \lambda(\vec{G}) + 1 \leq \lambda(\vec{G})$. From Property 13, we conclude that $\lambda(\vec{H}) = \lambda(\vec{G})$. ∎

A digraph $\vec{H}$ in $\Pi_k(G)$ is said maximal if no arc can be added to $\vec{H}$ so that the resulting digraph also belongs to $\Pi_k(G)$.

Property 15 *Let $\vec{H}$ be a maximal digraph in $\Pi_k(G)$, and let u and v be two vertices that are adjacent in G but not in H. Then $d_{\vec{H}}^-(u) = d_{\vec{H}}^-(v)$.*

Proof. If $d_{\vec{H}}^-(u) < d_{\vec{H}}^-(v)$, then we know from Property 12 and Corollary 14 that the graph obtained by adding the arc $u \to v$ in $\vec{H}$ also belongs to $\Pi_k(G)$. This contradicts the maximality of $\vec{H}$ in $\Pi_k(G)$, and we therefore have $d_{\vec{H}}^-(u) \geq d_{\vec{H}}^-(v)$. By permuting the roles of u and v, we also have $d_{\vec{H}}^-(u) \leq d_{\vec{H}}^-(v)$. ∎

Property 16 *Let $\vec{H}$ be a maximal digraph in $\Pi_k(G)$, and let $\vec{H}'$ be a digraph obtained from $\vec{H}$ by adding and orienting an edge that belongs to G but not to H. Then*

- *$\vec{H}'$ is circuit-free;*

- *$\lambda(\vec{H}') > k$.*

Proof. Let $u \to v$ be the arc that has been added to $\vec{H}$ to obtain $\vec{H}'$. We know from Property 15 that $d_{\vec{H}}^-(u) = d_{\vec{H}}^-(v)$. Hence, there is no path from v to u in $\vec{H}$, which means that $\vec{H}'$ is circuit-free. Since $\vec{H}$ is maximal in $\Pi_k(G)$, we therefore have $\lambda(\vec{H}') > k$. ∎

By removing some arcs from a digraph $\vec{G}$, one gets a digraph $\vec{H}$ with $\lambda(\vec{H}) \leq \lambda(\vec{G})$. The next Property indicates when $\lambda(\vec{H})$ is strictly smaller than $\lambda(\vec{G})$.

Property 17 *Let $\vec{H}$ be the digraph obtained by removing a subset F of arcs from an orientation $\vec{G}$ of G. Then $\lambda(\vec{H}) < \lambda(\vec{G})$ if and only if every longest path in $\vec{G}$ has at least one arc in F.*

Proof. If there is a longest path in $\vec{G}$ with no arc in F, then this path also exists in $\vec{H}$ which means that $\lambda(\vec{H}) = \lambda(\vec{G})$. Otherwise, all paths in $\vec{H}$ have a length strictly smaller than $\lambda(\vec{G})$. ∎

We close this section with a property that demonstrates a result similar to Properties 7 and 10, but for the *k-fixed partial orienting strategy*, where a neighbor is obtained by adding an arc and by removing some others so that the resulting digraph belongs to $\Pi_k(G)$. More precisely, for a digraph $\vec{H} \in \Pi_k(G)$, let $N_4(\vec{H})$ be the set of digraphs $\vec{H}'$ constructed from $\vec{H}$ as follows.

1. A maximal digraph $\vec{D}$ in $\Pi_k(G)$ is first obtained from $\vec{H}$ by adding arcs that create no circuits and no paths of length $> k$. Notice that $\vec{D} = \vec{H}$ if $\vec{H}$ is already maximal in $\Pi_k(G)$.

2. If $\vec{D} \in \Omega(G)$, then the neighbor $\vec{H}' \in N_4(\vec{H})$ of $\vec{H}$ is defined equal to $\vec{D}$.

   If $\vec{D} \notin \Omega(G)$, a digraph $\vec{D}'$ is built from $\vec{D}$ by adding and orienting an edge that belongs to G but not to D. We know from Property 16 that $\vec{D}'$ is circuit-free but contains paths of length $> k$. The neighbor $\vec{H}' \in N_4(\vec{H})$ of $\vec{H}$ is then obtained from $\vec{D}'$ by removing a minimal (inclusion-wise) set of arcs so that no path in $\vec{H}'$ has length $> k$.

Property 18 *Let G be a graph and let k be an integer larger than or equal to $\chi(G)$. For every $\vec{H} \in \Pi_k(G)$ there is a sequence $(\vec{H}_0, \cdots, \vec{H}_n)$ of digraphs such that $\vec{H}_0 = \vec{H}$, $\vec{H}_n \in \Omega(G)$, and each $\vec{H}_i$ belongs to $N_4(\vec{H}_{i-1})$.*

Proof. Let $\vec{G}^*$ be a circuit-free orientation of G with $\lambda(\vec{G}^*) \leq k$. Also, for a digraph $\vec{H} \in \Pi_k(G)$, let $B(\vec{H})$ denote the set of arcs in $\vec{H}$ that are oriented as in $\vec{G}^*$. Let $\vec{D}_i$ be a maximal digraph in $\Pi_k(G)$ obtained from $\vec{H}_i$ by adding arcs that create no circuits and no paths of length $> k$. We have $| B(\vec{D}_i) | \geq | B(\vec{H}_i) |$. If $\vec{D}_i \in \Omega(G)$ then define $\vec{H}_{i+1}$ equal to $\vec{D}_i$, else let $\vec{D}'_i$ denote the digraph obtained from $\vec{D}_i$ by adding an arc $u \to v$ of $\vec{G}^*$ between two non adjacent vertices u and v in $\vec{D}_i$. We know from Property 16 that $\vec{D}'_i$ has no circuits but has paths of length $> k$. These too long paths do not exist in $\vec{G}^*$ and can therefore be removed from $\vec{D}'_i$ by deleting a minimal (inclusion-wise) set of arcs $x \to y$ so that $\vec{G}^*$ contains the opposite arc $y \to x$. After all these removals, one gets a digraph $\vec{H}_{i+1}$ with $| B(\vec{H}_{i+1}) | = | B(\vec{D}'_i) | = | B(\vec{D}_i) | + 1 \geq | B(\vec{H}_i) | + 1$. One can repeat this process until one reaches a digraph $\vec{H}_n \in \Omega(G)$. This will occur in at most $| E(G) | - | B(\vec{H}_0) |$ steps since $\vec{H}_n = \vec{G}^*$ when $| B(\vec{H}_n) | = | E(G) |$. ∎

3 Four tabu search algorithms

Tabu search is one of the most famous local search techniques. It was introduced by Glover in 1986. A description of the method and its concepts can be found in [6]. For a solution s in the search space S, its neigbhorhood $N(s)$ is defined as the set of solutions $s' \in S$ obtained from s by performing a *local change m* on it. We denote $s' = s \oplus m$. A basic tabu search is described in Figure 4.

Choose an initial solution s; set $TL = \emptyset$ (tabu list); set $s^* = s$ (best solution)
Repeat the following until a stopping criterion is met
- Determine a best solution $s' \in N(s)$ such that either $s' = s \oplus m$ with $m \notin TL$ or s' is better than s^*
- If s' is better than s^* then set $s^* := s'$
- Set $s := s'$ and update TL

Figure 4 : Basic tabu search

In this section, we describe four tabu search algorithms for the graph coloring problem. Three of them use the *total orienting strategy*, and one uses the *k-fixed partial orienting strategy*. In order to describe these algorithms, it is sufficient to clearly define the search space S, the objective function to be minimized, the neighborhood of each solution, and the contents of the tabu list TL.

For a circuit-free digraph $\vec{G}$, we denote

- $W(\vec{G})$ its set of vertices on longest paths;
- $L(\vec{G})$ its set of arcs on longest paths;
- $A_{\vec{G}}^{-}(v)$ the set of arcs in $L(\vec{G})$ having vertex v as head, and $A_{\vec{G}}^{+}(v)$ the set of arcs in $L(\vec{G})$ having vertex v as tail (see Section 2);
- $\vec{L}_{\ell}(\vec{G})$ the partial digraph of $\vec{G}$ containing only those arcs $u \to v$ on longest paths such that $d_{\vec{G}}^{-}(u) = \ell$ and $d_{\vec{G}}^{-}(v) = \ell + 1$, where ℓ is an integer in $\{1, \cdots, \lambda(\vec{G}) - 1\}$ (see Section 2);
- $N_1(\vec{G})$ the set of digraphs obtained from $\vec{G}$ by changing the orientation of exactly one arc in $L(\vec{G})$;
- $N_2(\vec{G})$ the set of digraphs obtained from $\vec{G}$ by choosing a vertex v in $W(\vec{G})$ and by changing the orientation of all arcs in $A_{\vec{G}}^{-}(v)$, or of all arcs in $A_{\vec{G}}^{+}(v)$;
- $N_3(\vec{G})$ the set of digraphs obtained from $\vec{G}$ by choosing a number $\ell \in \{1, \cdots, \lambda(\vec{G}) - 1\}$ and by changing the orientation of all arcs in a subset of connected components of $\vec{L}_{\ell}(\vec{G})$.

We known from Properties 6, 9, and 11 that all digraphs in $N_i(\vec{G})$ ($i = 1, 2, 3$) are circuit-free. Moreover,

- $\lambda(\vec{H}) \leq \lambda(\vec{G}) + 2$ for all $\vec{H} \in N_1(\vec{G})$;
- $\lambda(\vec{H}) \leq \lambda(\vec{G}) + 1$ for all $\vec{H} \in N_2(\vec{G})$;
- $\lambda(\vec{H}) \leq \lambda(\vec{G})$ for all $\vec{H} \in N_3(\vec{G})$.

The first tabu search algorithm, called TABU_1, has $\Omega(G)$ as search space and uses neighborhood N_1. The objective is to determine an orientation $\vec{G}$ of G with minimum $\lambda(\vec{G})$. Since many orientations of G may have longest paths of the same length, we discriminate two orientations $\vec{G}$ and $\vec{H}$ with $\lambda(\vec{G}) = \lambda(\vec{H})$ by counting their number of arcs on longest paths. More precisely, a solution $\vec{G}$ is better than a solution $\vec{H}$ if and only if $\lambda(\vec{G}) < \lambda(\vec{H})$ or $\lambda(\vec{G}) = \lambda(\vec{H})$ and $| L(\vec{G}) | < | L(\vec{H}) |$. The tabu list contains edges with the meaning that it is forbidden to change their orientation. When performing a move from a solution $\vec{G}$ to a neighbor solution $\vec{H} \in N_1(\vec{G})$, the orientation of an arc $u \to v$ is changed and the edge (u, v) is introduced in the tabu list TL. We have proved in Property 7 that for every $\vec{G} \in \Omega(G)$ there is a sequence $(\vec{H}_0, \cdots, \vec{H}_n)$ of digraphs such that $\vec{H}_0 = \vec{G}$, $\lambda(\vec{H}_n) = \chi(G)$, and each $\vec{H}_i$ belongs to $N_1(\vec{H}_{i-1})$ ($i = 1, \cdots, n$).

The second tabu search algorithm, called TABU_2, also has $\Omega(G)$ as search space but uses neighborhood N_2. As for TABU_1, a solution $\vec{G}$ is better than a solution $\vec{H}$ if and only

if $\lambda(\vec{G}) < \lambda(\vec{H})$ or $\lambda(\vec{G}) = \lambda(\vec{H})$ and $\mid L(\vec{G}) \mid < \mid L(\vec{H}) \mid$. When a neighbor $\vec{H} \in N_2(\vec{G})$ of a solution $\vec{G}$ is obtained by changing the orientation of all arcs $u \to v$ in $A_{\vec{G}}^-(v)$, the pair $(v, +)$ is introduced in the tabu list TL, with the meaning that it is now forbidden for several iterations to change the orientation of an arc having v as tail. Similarly, when $\vec{H}$ is obtained from $\vec{G}$ by changing the orientation of all arcs $v \to w$ in $A_{\vec{G}}^+(v)$, the pair $(v, -)$ is introduced in TL in order to forbid the change of the orientation of an arc having v as head. We have proved in Property 10 that for every $\vec{G} \in \Omega(G)$ there is a sequence $(\vec{H}_0, \cdots, \vec{H}_n)$ of digraphs such that $\vec{H}_0 = \vec{G}$, $\lambda(\vec{H}_n) = \chi(G)$, and each $\vec{H}_i$ belongs to $N_2(\vec{H}_{i-1})$ $(i = 1, \cdots, n)$.

The third tabu search algorithm that uses the *total orienting strategy* is called TABU_3. As for TABU_1 and TABU_2, the solution space S is $\Omega(G)$, and a solution $\vec{G}$ is better than a solution $\vec{H}$ if and only if $\lambda(\vec{G}) < \lambda(\vec{H})$ or $\lambda(\vec{G}) = \lambda(\vec{H})$ and $\mid L(\vec{G}) \mid < \mid L(\vec{H}) \mid$. However, the neighborhood in this case is N_3. The tabu list TL contains edges with the meaning that it is forbidden to move from a solution $\vec{G}$ to a solution $\vec{H} \in N_3(\vec{G})$ if such a move requires to change the orientation of at least one edge in TL. When performing a move from a solution $\vec{G}$ to a neighbor solution $\vec{H} \in N_3(\vec{G})$, some edges change their orientation, and all these edges are introduced in the tabu list TL. As shown in the previous section, there are orientations $\vec{G} \in \Omega(G)$ for which there is no sequence $(\vec{H}_0, \cdots, \vec{H}_n)$ of digraphs with $\vec{H}_0 = \vec{G}$, $\lambda(\vec{H}_n) = \chi(G)$, and each $\vec{H}_i$ belongs to $N_3(\vec{H}_{i-1})$.

The fourth and last tabu search algorithm, called TABU_4, is based on the *k-fixed partial orienting strategy*. It works with a fixed integer k and tries to determine an orientation $\vec{G}$ of a given graph G so that $\lambda(\vec{G}) \leq k$. The search space S is $\Pi_k(G)$, and a solution is better than another one if and only if it contains more arcs. A neighbor $\vec{H}' \in N_4(\vec{H})$ of a solution $\vec{H}$ is obtained as follows (see also Section 2). A maximal digraph $\vec{D}$ in $\Pi_k(G)$ is first obtained from $\vec{H}$ by adding arcs that create no circuits and no paths of length $> k$. If $\vec{D} \in \Omega(G)$, then the neighbor $\vec{H}'$ of $\vec{H}$ is defined equal to $\vec{D}$ and is optimal since it contains $\mid E(G) \mid$ arcs. A new search can therefore be initiated with a smaller value of k. If $\vec{D} \notin \Omega(G)$, a digraph $\vec{D}'$ is built from $\vec{D}$ by adding and orienting an edge that belongs to G but not to D. We know from Property 16 that $\vec{D}'$ is circuit-free but contains paths of length $> k$. Finally, the neighbor $\vec{H}' \in N_4(\vec{H})$ of $\vec{H}$ is obtained from $\vec{D}'$ by removing a minimal (inclusion-wise) set of arcs so that no path in $\vec{H}'$ has length $> k$. The unique arc that is added to $\vec{D}$ to obtain $\vec{D}'$ is introduced in the tabu list TL. This list contains arcs that are not allowed to be removed for several iterations. We have proved in Property 18 that for every $\vec{H} \in \Pi_k(G)$ there is a sequence $(\vec{H}_0, \cdots, \vec{H}_n)$ of digraphs such that $\vec{H}_0 = \vec{H}$, $\vec{H}_n \in \Omega(G)$, and each $\vec{H}_i$ belongs to $N_4(\vec{H}_{i-1})$.

4   Computational experiments and conclusions

The objective of the computational experiments is to compare the four algorithms described in the last section. More specifically, we would like to have indications about the relative efficiency of the different neighborhoods. Hence, the computational experiments are not meant to be exhaustive, but rather indicative of the behavior of the algorithms. These results are

nonetheless very useful, as they will help to orient future research on the development of more elaborate algorithms.

To test the four algorithms on a graph G, we consider the circuit-free orientation $\vec{G}_0$ of G obtained by labelling the vertices of G from 1 to $| V(G) |$, and by orienting each edge (u, v) from u to v if and only if u has a smaller label than v. TABU_1, TABU_2 and TABU_3 are all run for 100,000 iterations starting from $\vec{G}_0$, and we report the smallest length of the longest path ever encountered during the search.

The test procedure is a little bit different for TABU_4. We first fix $k = \lambda(\vec{G}_0) - 1$ and construct a digraph $\vec{H}$ by removing from $\vec{G}_0$ a minimal (inclusion-wise) subset of arcs so that $\lambda(\vec{H}) = k$. If TABU_4 is able to produce a digraph $\vec{G} \in \Omega(G)$ in less than 100,000 iterations, then we decrease k by one unit, we build a new initial solution $\vec{H}$ by removing from $\vec{G}$ a minimal (inclusion-wise) subset of arcs so that $\lambda(\vec{H}) = k$, we reset the iteration counter to zero, and start a new search. We report the smallest value k for which 100,000 iterations were sufficient for TABU_4 to produce a solution $\vec{H} \in \Omega(G)$ with $\lambda(\vec{H}) = k$.

We compare the four tabu search algorithms on ten random graphs with edge density $d = 0.5$. These graphs are obtained by linking a pair of vertices by an edge with probability 0.5, independently for each pair. It is known in the graph coloring community that random graphs with $d = 0.5$ are hard to color. Five of our instances have 50 vertices and are labelled $G_{50,i}$ $(i = 1, \cdots, 5)$ while the five other graphs have 100 vertices and are labelled $G_{100,i}$ $(i = 1, \cdots, 5)$. For comparison, we also indicate the number of colors used by $Tabucol$, which is considered as one of the most simple and efficient tabu search coloring algorithm [3]. All results are given in Table 1.

It clearly appears that TABU_2 is not competitive at all. TABU_1 and TABU_4 give similar results, with a slight advantage for TABU_4. The best tabu search with an edge orienting strategy is indisputably TABU_3 which is even competitive with $Tabucol$ on graphs with 50 vertices. Notice that TABU_3 is the unique tabu search algorithm studied in this paper for which the search space is not connected (see the example on Figure 3).

Ten graphs is of course a small test set for comparing heuristic methods. We think however that these limited experiments give a clear indication on which strategies seem to be the most promising. For the *total orienting strategy*, the reversal of arcs on longest paths between successive colors clearly gives better results than the reversal of a single arc on a longest path. Also, the reversal of a set of arcs on longest paths having all the same tail or the same head does not seem to be an efficient strategy. The unique tabu search algorithm that uses the *k-fixed partial orienting strategy* is TABU_4, and we have shown that it is ranked second among the four tested algorithms.

More experiments of course still have to be performed, and the combination of an edge orienting strategy with a more classical graph coloring algorithm would also be an interesting research subject. This will be the topic of future research based on the theoretical and experimental results contained in this paper.

Graph	TABU_1	TABU_2	TABU_3	TABU_4	Tabucol
$G_{50,1}$	13	19	10	12	10
$G_{50,2}$	13	16	9	13	9
$G_{50,3}$	14	16	10	13	10
$G_{50,4}$	13	14	9	12	9
$G_{50,5}$	12	15	9	12	9
$G_{100,1}$	27	34	16	25	15
$G_{100,2}$	28	38	16	25	16
$G_{100,3}$	27	36	16	25	15
$G_{100,4}$	27	32	15	25	15
$G_{100,5}$	26	34	16	26	15

Table 1: Comparison of five tabu search algorithms on ten graphs.

References

[1] BARBOSA, V.C., ASSIS, C.A.G., DO NASCIMENTO, J.O., Two Novel Evolutionary Formulations of the Graph Coloring Problem, *Journal of Combinatorial Optimization* 8, 2004, 41-63

[2] BROWN, J.R., Chromatic Scheduling and the Chromatic Number Problem, *Management Science* 19/4, 1972, 456-463

[3] GALINIER, P., HERTZ, A., A Survey of Local Search Methods for Graph Coloring, to appear in *Computers & Operations Research*, 2006

[4] GALLAI, T., On Directed Paths and Circuits, in: P. Erdös, G. Katobna (Eds.), *Theory of Graphs*, Tihany, Academic Press, New York, 1968, 115-118

[5] GAREY, M.R., JOHNSON, D.S., Computers and Intractability : A Guide to the Theory of NP-Completeness, W.H. Freman and Company, NY, 1979

[6] GLOVER, F., LAGUNA, M. (eds.), *Tabu Search*, Kluwer Academic Publishers, 1997

[7] HERRMANN, F., HERTZ, A., Finding the Chromatic Number by Means of Critical Graphs, *ACM Journal of Experimental Algorithmics* 7/10, 2002, 1-9

[8] KUBALE, M., JACKOWSKI, B., A Generalized Implicit Enumeration Algorithm for Graph Coloring, *Communications of the ACM* 28/4, 1985, 412-418

[9] MEHROTRA, A., TRICK, M.A., A Column Generation Approach for Exact Graph Coloring, *INFORMS Journal on Computing* 8, 1996, 344-354

[10] PEEMÖLLER, J., A Correction to Brélaz's Modification of Brown's Coloring Algorithm, *Communications of the ACM* 26/8, 1983, 593-597

[11] ROY, B., Nombre Chromatique et Plus Longs Chemins d'un Graphe, *Revue AFIRO* 1, 1967, 127-132

[12] VAN LAARHOVEN, P.J.M., AARTS, E.H.L., LENSTRA, J.K., Job-Shop Scheduling by Simulated Annealing, *Operations Research* 40/1, 1992, 113-125

[13] VITAVER, L.M., Determination of Minimal Coloring of Vertices of a Graph by Means of Boolean Powers of the Incidence Matrix, *Dokl. Akad. Nauk. SSSR147*, 1962, 758-759 (in Russian)