Minimum Weighted Feedback Vertex Set on Diamonds

F.Carrabs [a], R.Cerulli [a], M.Gentili [a], G.Parlato [b]

[a] *Dipartimento di Matematica ed Informatica Università di Salerno, Via Ponte Don Melillo, 84084, Fisciano (SA), Italy*

[b] *Dipartimento di Informatica ed Applicazioni Università di Salerno, Via S. Allende, 84081, Baronissi (SA), Italy*

Abstract

Given a vertex weighted graph G, a minimum Weighted Feedback Vertex Set (MWFVS) is a subset $F \subseteq V$ of vertices of minimum weight such that each cycle in G contains at least one vertex in F. The MWFVS on general graph is known to be NP-hard. In this paper we introduce a new class of graphs, namely the *diamond* graphs, and give a linear time algorithm to solve MWFVS on it. We will discuss, moreover, how this result could be used to effectively improve the approximated solution of any known heuristic to solve MWFVS on a general graph.

Key words: Feedback Vertex Set, Dynamic Programming, Diamond Graphs

1 Introduction

Given a vertex weighted graph G, a minimum Weighted Feedback Vertex Set (MWFVS) is a subset $F \subseteq V$ of vertices of minimum weight such that each cycle in G contains at least one vertex in F. The MWFVS on general graph is known to be NP-hard. However, a large literature shows that it becomes polynomial when addressed on specific classes of graph: interval graph (1), cocomparability graph (2), AT-free graph, among others. In this paper we introduce a new class of graphs, namely the *diamond* graphs, and give a linear time algorithm to solve MWFVS on it. We will discuss, moreover, how this result could be used to effectively improve an approximated solution on a general graph, that is the object of our further research. Section 2 introduces

Email addresses: carrabs@dia.unisa.it (F.Carrabs), raffaele@unisa.it (R.Cerulli), mgentili@unisa.it (M.Gentili), parlato@dia.unisa.it (G.Parlato).

the needed notation and the diamond graph class. Section 3 contains the description of our linear algorithm based on dynamic programming to optimally solve MWFVS on diamonds. Further research is discussed in Section 4.

2 The class of diamond graphs

In this section we give some basic concepts used in this paper. Let $G = (V, E, w)$ be a weighted graph, where V is the set of vertices, $E \subseteq V \times V$ is the set of edges and $w : V \rightarrow \Re^+$ is a weight function which associates a positive real number to each vertex of G. Given a weighted graph $G = (V, E, w)$ and a subset X of V, we define $G - X = (V \setminus X, ((V \setminus X) \times (V \setminus X)) \cap E, w')$, where w' is the restriction of w on the domain $V \setminus X$.

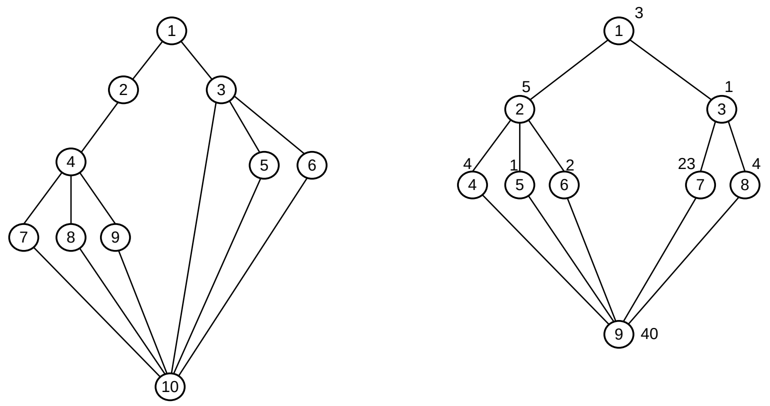
A *tree* is an acyclic, connected and undirected graph. Let T be a tree rooted in r. Given a vertex u in T, we denote as C_u the set of children of u in T. The *height* of a vertex u in T, denoted by $h(u)$, is recursively defined as follows. If u is a leaf then $h(u) = 0$, otherwise $h(u) = \max_{v \in C_u}\{h(v)\} + 1$. We define the height of a tree T as the height of its root (i.e $h(T) = h(r)$). Given a vertex $u \in T$, the *subtree* of T rooted in u is the subgraph of T induced by the set of vertices constituted by u and its descendents in T. In the following we denote by T_u the subtree of T rooted in u.

Now we introduce a new class of graphs, namely the *Diamond* graphs.

Definition 2.1 *A* ***diamond*** $D = (V, E, w)$ *with apices* r *and* z *(with* $r, z \in V$*), is a weighted graph where (i) each* $v \in V$ *is included in at least a simple path between* r *and* z *and (ii)* $D - \{z\}$ *is a tree.*

We call the two vertices r and z of D the *upper* and *lower* apex of D, respectively. Given a diamond D with apices r and z, we refer to the tree $D - \{z\}$ rooted in r as T. Let us denote as D_u the subdiamond of D, with apices u and z, induced by vertices of T_u and z. The *height* of diamond D_u is given by the height of T_u (see Figure 1(a)).

Given a diamond D with apices r and z, we can see that it is formed by the subdiamonds D_{r_i}, with $r_i \in C_r$, having the common lower apex z and upper apex r_i.



(a) (b)

Fig. 1. *(a) A diamond with apices* $r = 1$ *and* $z = 10$. *The height of this diamond is* 3 *and the associated tree* T *is* $D - \{z\}$. *Note that, the subgraph* D_2 *(with* $V_2 = \{2, 4, 7, 8, 9, 10\}$*) is a subdiamond of height 2 with* $r = 2$ *and* $z = 10$.*(b) A diamond with apices* $r = 1$ *and* $z = 9$. *A minimum FVS (that does not contain vertex* z*) is* $\hat{F} = \{3, 5, 6\}$ *and since* $W(\hat{F}) = 4 < w(z) = 40$ *we have that* $F^* = \hat{F}$. *Note that in* $D - \hat{F}$ *there is a simple path between* r *and* z. *Setting* $w(2) = 2$, *the set* $\hat{F} = \{2, 3\}$ *with* $W(\hat{F}) = 3$ *is the optimum FVS. In this case* $D - \hat{F}$ *does not contain a path between* r *and* z

3 Our resolution algorithm

In this section, we propose a linear algorithm, based on dynamic programming, to solve the MWFVS problem on diamonds. In the rest of this section we consider a diamond D having r and z as its upper and lower apices. Note that by definition the set $F = \{z\}$ is an FVS of D. Hence, an optimum solution F^* of MWFVS on D is such that either $F^* = \{z\}$ or, $z \notin F^*$ and $W(F^*) \leq w(z)$. Hence, the MWFVS problem on a diamond can be solved first finding a minimum FVS $\hat{F}$ that does not contain vertex z and then comparing $W(\hat{F})$ with $w(z)$. Let us notice that in $D - \hat{F}$ either there exists a simple path between r and z or it does not exist (see Figure 1(b)). We prove the above observation by the following proposition.

Proposition 3.1 *Given a diamond* D *with apices* r *and* z, *if* $F \subseteq V$ *is a (minimum) FVS of* D, *then there exists at most one path between* r *and* z *in* $D - F$.

Let us define now two new problems on a diamond D, related to MWFVS that will be useful to solve MWFVS on D.

Path Problem: find a subset $F^+ \subseteq V \setminus \{z\}$ of minimum weight such that (i) $D - F$ does not contain cycles, and, (ii) there exist exactly a path in $D - F$ between r and z.

NoPath Problem: find a subset $F^- \subseteq V \setminus \{z\}$ of minimum weight such that (i) $D - F$ does not contain cycles, and, (ii) there does not exist a path in $D - F$ between r and z.

From the above observations the minimum $\hat{F}$ on D is either F^+ or F^-. Therefore, the optimum solution F^* on D is such that:

$$W(F^*) = \min\{w(z), W(F^+), W(F^-)\}$$

and, then we have either $F^* = \{z\}$ or $F^* = F^+$ or $F^* = F^-$.

3.1 Optimal Substructure and Recursion Rules

In this section, we conjunctly characterize the structure of an optimal solution for both *Path* and *NoPath* problems. We recall that a problem has an optimal substructure if an optimal solution to the problem contains within it optimal solutions to subproblems. Both *Path* and *NoPath* problems have an optimal substructure property. We pick as our subproblems the problems of determining the value of solution to *Path* and *NoPath* problems on the subdiamonds D_u with $u \in V \setminus \{z\}$. We will denote by F_u^+ and F_u^- the optimal solutions of the *Path* and *NoPath* problem, respectively, on D_u. Now, we prove that the optimal solution F_u^+ (F_u^-) contains, for each $u_i \in C_u$, either $F_{u_i}^+$ or $F_{u_i}^-$. .

Proposition 3.2 *Given the optimum solution F_u^+ on D_u, then each set $F_{u_i} = F_u^+ \cap V_{u_i}$ is an optimal solution to either Path problem or NoPath problem on D_{u_i}.*

Proposition 3.3 *Given the optimum solution F_u^- on D_u, then each set $F_{u_i} = F_u^- \cap V_{u_i}$ is an optimal solution to either Path problem or NoPath problem on D_{u_i}.*

Now, we describe the recursion rules to obtain the values of the optimum sets F_u^+ and F_u^-. We define the $W(F_u^+)$ and $W(F_u^-)$ recursively as follows.

Optimum Solution to Path Problem: $W(F_u^+)$

We distinguish two cases according to the height $h(u)$:

- $h(u) = 0$
  Trivially $W(F_u^+) = 0$.

- $h(u) > 0$
  If $(u, z) \in E_u$, then since $u \notin F_u^+$, $G_u - F_u^+$ has surely a path between u and z composed by the edge (u, z). Thus, to avoid cycles in $G_u - F_u^+$, the set F_u^+ is obtained by the union of F_x^-, for each $x \in C_u$ and $W(F_u^+) = \sum_{x \in C_u} W(F_x^-)$.

Instead, when $(u, z) \notin E_u$, in order to have a path from u to z in $G_u - F_u^+$, the optimal set is obtained by the minimum weight union of exactly one set F_x^+, for some $x \in C_u$, and by $\bigcup_{y \in C_u - \{x\}} F_y^-$. Therefore, we have the following:

$$W(F_u^+) = \min_{x \in C_u} \{W(F_x^+) + \sum_{y \in C_u - \{x\}} W(F_y^-)\}$$

By applying a similar reasoning we can derive the following recursion rules for the *NoPath* problem.

Optimum Solution to NoPath Problem: $W(F_u^-)$

- $h(u) = 0$
  $F_u^- = \{u\}$ and $W(F_u^-) = w(u)$.
- $h(u) > 0$

$$W(F_u^-) = \min \left\{ w(u) + \sum_{x \in C_u} \min \left\{ W(F_x^-), W(F_x^+) \right\}, \sum_{x \in C_u} W(F_x^-) \right\}$$

4 Conclusion and Further Research

We studied the Weighted Feedback Vertex Set on a special class of graph: the diamonds graph. We showed a linear time algorithm to solve the problem on this graph class. Further research is focused on the study of the larger class of multidiamond graphs (diamonds with multi-upper and lower apices). In addition, our purpose is to use the linearity of the MWFVS on this class of graph to try to improve an approximated solution of MWFVS on general graph.

References

[1] C.L.Lu, C.Y.Tang, *A linear-time algorithm for the weighted feedback vertex set problem on interval graphs.* Information Processing Letters 61 (1997) 107-111.

[2] M.S. M.S. Chang, Y.D. Liang: *Minimum feedback vertex sets in cocomparability graphs and convex bipartite graphs*, em Acta Informatica Vol.34 (1997), 337-346 Thomas Erlebach, Stamatis Stematis Stefanakos *Wavelength Conversion in Network of Bounded Treewidth* Tik Report Nr. 132, April 2002