RAIRO Operations Research
RAIRO Oper. Res. **41** (2007) 275–287
DOI: 10.1051/ro:2007024

CLIQUE PARTITIONING OF INTERVAL GRAPHS WITH SUBMODULAR COSTS ON THE CLIQUES

Dion Gijswijt[1], Vincent Jost[2] and Maurice Queyranne[2]

Abstract. Given a graph $G = (V, E)$ and a "cost function" $f : 2^V \rightarrow \mathbb{R}$ (provided by an oracle), the problem [PCliqW] consists in finding a partition into cliques of $V(G)$ of minimum cost. Here, the cost of a partition is the sum of the costs of the cliques in the partition. We provide a polynomial time dynamic program for the case where G is an interval graph and f belongs to a subclass of submodular set functions, which we call "value-polymatroidal". This provides a common solution for various generalizations of the coloring problem in co-interval graphs such as max-coloring, "Greene-Kleitman's dual", probabilist coloring and chromatic entropy. In the last two cases, this is the first polytime algorithm for co-interval graphs. In contrast, NP-hardness of related problems is discussed. We also describe an ILP formulation for [PCliqW] which gives a common polyhedral framework to express min-max relations such as $\overline{\chi} = \alpha$ for perfect graphs and the polymatroid intersection theorem. This approach allows to provide a min-max formula for [PCliqW] if G is the line-graph of a bipartite graph and f is submodular. However, this approach fails to provide a min-max relation for [PCliqW] if G is an interval graphs and f is value-polymatroidal.

Keywords. Partition into cliques, Interval graphs, Circular arc graphs, Max-coloring, Probabilist coloring, Chromatic entropy, Partial q-coloring, Batch-scheduling, Submodular functions, Bipartite matchings, Split graphs.

Mathematics Subject Classification. 90C27, 05C15.

Received November 15, 2006. Accepted December 21, 2006.

[1] Dep. of Operations Research, EGRES, Eötvös Lorand University, Pázmány Peter Setany. 1/C, 1117 Budapest, Hungary; dion.gijswijt@gmail.com

[2] CNRS, laboratoire Leibniz-IMAG, 46 avenue Félix Viallet, 38031 Grenoble Cedex, France; vincent.jost@imag.fr

1. Introduction

Let $G = (V, E)$ be a simple graph. In the following, a *clique* of G refers to a non-empty subset of vertices inducing a complete subgraph (not necessarily maximal with this property). Let $\mathcal{C}(G)$ denote the set of cliques of G. A partition into cliques of G is a partition $\mathcal{Q} = (K_1, \ldots, K_k)$ of $V(G)$, where $K_1, \ldots, K_k \in \mathcal{C}(G)$. In other words it is a coloring of $\overline{G}$, the complementary graph of G. Let $\mathcal{P}(G)$ denote the set of all partitions into cliques of G. A classical problem consists in determining $\overline{\chi}(G)$, the minimum number of cliques necessary to partition G. In several applications however (see Sect. 3), there is a *cost* $f(C)$ associated to every clique $C \in \mathcal{C}(G)$, and we are interested in partitioning G into cliques, minimizing the sum of the costs of the cliques in the partition. Let $\overline{\chi}(G, f)$ denote this minimum:

$$\overline{\chi}(G, f) := \min_{\mathcal{Q} \in \mathcal{P}(G)} \sum_{K \in \mathcal{Q}} f(K). \tag{1}$$

In order to describe some properties of f, one may assume that f is not only defined on cliques but is a *set function on* $\mathbf{V}$, that is $f : 2^V \to \mathbb{R}$. This has no consequences for the definitions of $\overline{\chi}(G, f)$ and [PCliqW] below. Notice that if $f(C) = 1$ for all cliques C, we get the classical problem of coloring $\overline{G}$ and we have $\overline{\chi}(G, \mathbf{1}) = \overline{\chi}(G)$. Determining $\overline{\chi}(G, f)$ is therefore an NP-hard problem. Moreover, since $|\mathcal{C}(G)|$ is usually exponential in $|V|$ (the complete graph K_n on n vertices has $|\mathcal{C}(K_n)| = 2^n$), encoding f itself raises complexity issues. In several applications however, both G and f have structural properties that allow to solve problem [PCliqW] in time polynomial in $|V|$.

[PCliqW] Partition into cliques with weights
INPUT : A graph $G = (V, E)$ and a value oracle, providing $f(K)$ in constant time for each $K \in \mathcal{C}(G)$.
OUTPUT : A partition into cliques of cost $\overline{\chi}(G, f)$.

[PCliqW] can also be described in terms of batch scheduling with compatibility graphs [12]. In this terminology (see [4] for batch scheduling problems not involving compatibility graphs and [16] for a classification of chromatic scheduling problems), each clique of a partition into cliques of G is called a *batch*. The operating time of a batch K is then $f(K)$ and our objective is to minimize the makespan $\mathrm{C}_{\max}$ (whence the batches are ordered arbitrarily on the batch machine). Talking about cliques and batches allows to distinguish easily between cliques of G and cliques in a partition of $V(G)$. Two famous polytime cases of [PCliqW] are when

- G is perfect and $f \equiv 1$ [17];
- G is complete and f is a submodular set function [17]

Our solution for [PCliqW] for interval graphs and value-polymatroidal functions can be seen as a compromise between these two classical cases. Moreover, [PCliqW] enjoys a simple min-max formula in both cases [17] ($\overline{\chi}(G) = \alpha(G)$ in the first case and "Dilworth's truncation" in the second). One could therefore expect a common generalized min-max formula to hold in other cases for which [PCliqW] is polynomial. We deal with this issue in Section 7.

In Section 2, we define polymatroid rank functions and motivate the definition of value-polymatroidal set functions in the context of [PCliqW]. In Section 3, we provide examples of value-polymatroidal set functions. In Section 4, we discuss value-polymatroidal functions whose values $f(U)$ depend only on the size $|U|$. In Section 5, we provide a dynamic program which solves [PCliqW] for interval graphs in polytime if f is value-polymatroidal. The algorithm extends to the minimum cost partition problem for circular arc graphs, when we only consider cliques in which the arcs share a common point. As a counterpart, we mention NP-hardness of [PCliqW] for interval graphs if f is only assumed to be polymatroidal [2]. In Section 6, we discuss NP-hardness of [PCliqW] on split graphs for subclasses of value-polymatroidal set functions. In Section 7, we deal with some polyhedral issues and provide a min-max formula for [PCliqW] in line-graphs of bipartite graphs.

2. Value-polymatroidal set functions

A set function $f : \mathcal{P}(V) \to \mathbb{R}$ is *submodular* if it satisfies one of the following equivalent properties [17]:

$$f(S \cup T) + f(S \cap T) \leq f(S) + f(T) \quad \text{for all } S, T \subseteq V, \tag{2}$$

$$f(S + u) + f(T) \leq f(S) + f(T + u) \quad \text{for all } T \subseteq S \subseteq V \text{ and } u \in V \backslash S, \tag{3}$$

$$f(S + u + v) + f(S) \leq f(S + u) + f(S + v) \quad \text{for all } S \subseteq V \text{ and } u, v \in V \backslash S. \tag{4}$$

A set function f is *non-negative* if all its values are, *non-decreasing* if $S \subseteq T \Longrightarrow f(S) \leq f(T)$, *subcardinal* if $f(U) \leq |U|$ for all $U \subseteq V$. A *polymatroid* rank function is a submodular, non-negative, non-decreasing set function such that $f(\emptyset) = 0$. A *matroid* rank function is a subcardinal, integral polymatroid rank function.

In some graph classes, submodularity of f is enough to ensure polynomiality of [PCliqW] (see Sect. 7 and [16]). Although submodularity is not sufficient for interval graphs (see Th. 5.5), a stronger exchange property will do. We say that f is a *value-polymatroidal* set function if $f(\emptyset) = 0$, f is non-decreasing and for every S and T subsets of V such that $f(S) \geq f(T)$ and every $u \in V \backslash (T \cup S)$, we have

$$f(S + u) + f(T) \leq f(S) + f(T + u). \tag{5}$$

Proposition 2.1. *Every value-polymatroidal set function is a polymatroid rank function.*

Proof. Let f be value-polymatroidal. Since f is non-decreasing, we have $f(S) \geq f(T)$ for every $T \subseteq S \subseteq V$ and therefore $f(S + u) + f(T) \leq f(S) + f(T + u)$ for every $u \in V \backslash S$. □

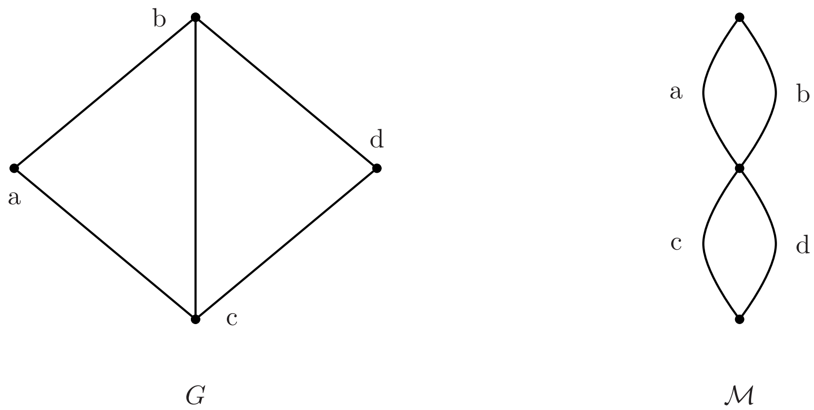

FIGURE 1. A graph G and a graphic matroid $\mathcal{M}$ (whose rank function is not value-polymatroidal) such that $\overline{\chi}(G, r(\mathcal{M})) = 2 = r(\{a, b\}) + r(\{c, d\})$. No optimal partition contains a maximal clique of G.

By a *maximal clique*, we mean a clique maximal for inclusion (not necessarily for cardinality). The main motivation behind the definition of value-polymatroidal set functions is given by the following proposition.

Proposition 2.2. *For any graph G and any value-polymatroidal set function f on $V(G)$, there is a partition $\mathcal{Q}$ of cost $\overline{\chi}(G, f)$ in which one of the cliques in $\mathcal{Q}$ is a maximal clique of G.*

Proof. Let $\mathcal{Q}$ be a minimum cost partition of G and choose any clique $K \in \mathcal{Q}$, such that $f(K) \geq f(T)$ for all $T \in \mathcal{Q}$. If K is not a maximal clique of G, there exists some $t \in V \backslash K$ such that $K + t$ is a clique in G. Now, t belongs to some $T \in \mathcal{Q} - K$. Since f is non-decreasing, $f(K) \geq f(T) \geq f(T - t)$. Since f is value-polymatroidal, $f(K+t) + f(T-t) \leq f(K) + f(T)$. Repeat the process until K becomes a maximal clique of G. □

In general, rank functions of (poly)matroids are not value-polymatroidal, and the conclusion of Proposition 2.2 doesn't hold as shown in Figure 1.

3. Examples of value-polymatroidal set functions

In this section we mention some (coloring) problems that have been studied in the literature, and that amount to solving [PCliqW] for special subclasses of value-polymatroidal set functions. These problems are often formulated is terms of finding a minimum cost partition into stable sets, which is equivalent to [PCliqW] by taking the complementary graph.

Maximum. Let $p : V \to \mathbb{R}_+$ and define

$$f(U) := \max_{u \in U} p(u) \tag{6}$$

for any $U \subseteq V$. Then f is value-polymatroidal. Indeed, let $S, T \subseteq V$ with $f(S) \geq f(T)$, and let $u \in V \backslash (S \cup T)$. Then, since $p(s) = f(S) \geq f(T) = p(t)$ for some $s \in S$ and $t \in T$, we have

$$f(S+u)+f(T) = \max\{p(s), p(u)\}+p(t) \leq p(s)+\max\{p(t), p(u)\} = f(S)+f(T+u).$$

A set function arising as in (6) is called a *max-batch cost function.* When restricted to max-batch cost functions, the corresponding problem of finding a minimum cost partition into stable sets is called [max-coloring] and is strongly-NP-hard for split graphs [3, 8], for bipartite graphs [8] and for interval graphs [11]. However, [max-coloring] is polynomial for P_4-free graphs [8] as well as for co-interval graphs [2, 9, 12].

Independent probabilities. Let $q : V \to [0, 1]$ and for $U \subseteq V$, let

$$f(U) := 1 - \Pi_{u \in U} q(u). \tag{7}$$

Let $S, T \subseteq V$ with $f(S) \geq f(T)$, and $u \in V \backslash (S \cup T)$. Write $f(S) = 1 - \sigma$ and $f(T) = 1 - \tau$ (so $\sigma \leq \tau$). Then

$$\begin{aligned} f(S) + f(T+u) &= (1-\sigma) + (1 - q(u)\tau) \\ &\geq (1 - q(u)\sigma) + (1-\tau) = f(S+u) + f(T). \end{aligned}$$

Hence f is value-polymatroidal. A set function arising as in (7) is a *probabilistic cost function.* Transitive references for applications of probabilist optimization can be found in [7].

When restricted to probabilistic cost functions, [PCliqW] is strongly NP-hard in split graphs [7]. The corresponding problem of partitioning into stable sets is called [probabilist coloring].

Chromatic Entropy. Let $p : V \to [0, 1]$ and for $U \subseteq V$, let

$$c_U := \sum_{u \in U} p(u) \tag{8}$$

$$f'(U) := -c_U \log(c_U). \tag{9}$$

If $c_V = 1$, f' is a *chromatic entropy* cost function. Although f' is not value-polymatroidal (it is not non-decreasing), the function $f := f' + c$ *is* value-polymatroidal as can be derived from the concavity of the function $x \mapsto x - x\log(x)$ [1]. Since for any partition $V = K_1 \cup \cdots \cup K_k$ of V into cliques, we have $\sum_i f(K_i) = c(V) + \sum_i f'(K_i)$, the two functions f' and f yield the same optimal partitions.

The corresponding problem of partitioning into stable sets is called [chromatic entropy] [1, 6] and is strongly NP-hard for interval graphs [6].

Uniform matroid and Partial q-coloring. Let $q \in \mathbb{N}$ and let

$$f(U) := \min\{q, |U|\}. \tag{10}$$

Then f is value-polymatroidal, and the proof is left as an exercise since a more general statement is given with the next example. Functions arising this way are exactly the rank functions of uniform matroids. [PCliqW] with such a cost function arises in Greene-Kleitman's min-max relations stating that for any (co)-comparability graph G and any integer q, the maximum cardinality $\alpha_q(G)$ of the union of q stable sets of G satisfies $\alpha_q(G) = \overline{\chi}(G, f)$ (see [5] and [17], Sects. 14.6 and 14.7 on unions of chains and antichains in posets and Sect. 66.5*e* on "k-perfect" graphs for more details and references).
Size-defined concave. Assume that $f(\emptyset) = 0$ and that

$$f(U) := \psi(|U|) \tag{11}$$

for some $\psi : \mathbb{N} \to \mathbb{R}_+$. Then f is value-polymatroidal if and only if f is the rank of a polymatroid and also if and only if ψ has a non-decreasing concave extension on the real segment $[0, |V|]$ (see Sect. 4). The rank function of a uniform matroid is a special case.

4. Size-defined submodular set functions

In this section, we notice that if $f(U)$ only depends on $|U|$, then polymatroid ranks coincide with value-polymatroidal functions. Let $[a..b]$ denote the set of integers in the interval $[a, b]$. A set function f on V is *size-defined* if there exists a function $\psi : [0..|V|] \to \mathbb{R}$ such that $f(U) = \psi(|U|)$. The function ψ is then the *compact representation* of f. Recall that a function $f : [a, b] \to \mathbb{R}$ *is concave* if for all $c, d \in [a, b]$ we have $f(c) + f(d) \leq 2f((c + d)/2)$

Theorem 4.1. *Let f be a size-defined, non-decreasing set function such that $f(\emptyset) = 0$ and ψ be the compact representation of f. The following are equivalent:*

(i) f is value-polymatroidal
(ii) f is a polymatroid rank function
(iii) $2\psi(i) \geq \psi(i-1) + \psi(i+1)$ for all $i \in [1..|V| - 1]$
(iv) $\psi(i+1) - \psi(i) \geq \psi(j+1) - \psi(j)$ for all $i, j \in [0..|V| - 1]$, with $i < j$
(v) $\exists\, \widehat{\psi} : [0, |V|] \to \mathbb{R}$ concave such that $\psi(i) = \widehat{\psi}(i)$ for $i \in [0..|V|]$

Proof. (i) $\Longrightarrow$ (ii): Proposition 2.1
(ii) $\Longrightarrow$ (iii): Use definition (4) of polymatroids with $|S| = i - 1$.
(iii) $\Longrightarrow$ (iv): By induction on $j - i$. The case $j - i = 1$ being exactly iii). Adding $\psi(i+1) - \psi(i) \geq \psi(j+1) - \psi(j)$ and $2\psi(j+1) \geq \psi(j) + \psi(j+2)$ gives $\psi(i+1) - \psi(i) \geq \psi(j+2) - \psi(j+1)$.
(iv) $\Longrightarrow$ (i): For $S, T \subseteq V$, since f is size-defined and non-decreasing,

$$f(S) \geq f(T) \Longleftrightarrow \psi(|S|) \geq \psi(|T|) \Longleftrightarrow |S| \geq |T|$$

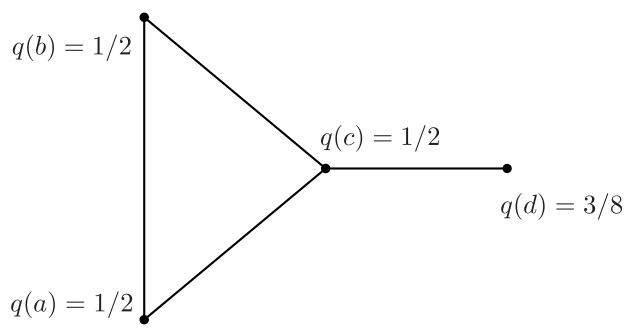

FIGURE 2. Let f be the probabilist cost defined by p. Vertex d has maximum cost $f(\{d\}) = 1 - q(d) = 5/8$. However, in an optimal partition, vertex d cannot be placed in a maximal clique since $25/16 = f(\{a,b\}) + f(\{c,d\}) > \overline{\chi}(G,f) = f(\{a,b,c\}) + f(\{d\}) = 12/8$.

Applying (iv) to $j = |S|$ and $i = |T|$ gives (i).
(v) $\Longrightarrow$ (iii): Apply the concavity condition to $c = i - 1$ and $d = i + 1$.
(iii) $\Longrightarrow$ (v): Take $\widehat{\psi}$ as the piecewise linear interpolation of f (for any $x \in [0..|V|]$, $\widehat{\psi}(x) := \lambda f(\lfloor x \rfloor) + (1-\lambda) f(\lceil x \rceil)$ for $\lambda := x - \lfloor x \rfloor$). One can check that the subgradient of $-\widehat{\psi}$ is nondecreasing. □

5. Partition into cliques in interval and circular arc graphs

A graph $G = (V, E)$ is an *interval graph* [13, 17] if there exists a set $\{\phi(v) \mid v \in V\}$ of closed intervals on the real line, such that two vertices u and v are adjacent in G if and only if the two corresponding intervals $\phi(u)$ and $\phi(v)$ have nonempty intersection. Observe that any maximal clique K in G is of the form $\{v \in V \mid t \in \phi(v)\}$ for some endpoint t of one of the intervals.

In [2, 9, 12], [PCliqW] is solved in polytime for interval graphs and max-batch cost functions. These algorithms use the fact that there exists an optimal solution in which a vertex of maximum cost is contained in a batch inducing a maximal clique. Based on this fact, a dynamic program is proposed. This fact is no longer true for value-polymatroidal costs as shown by the example in Figure 2. Nonetheless, based on Lemma 5.2, we describe a generalization of the algorithm proposed in [12], which provides an optimal solution for any value-polymatroidal cost function.

Theorem 5.1. *For any interval graph $G = (V, E)$ and any value-polymatroidal set function f on V given by a value oracle, we can compute a partition into cliques of G of cost $\overline{\chi}(G, f)$ in time $O(n^3)$.*

Proof. Let $\{I_i = [a_i, b_i]\}_{i=1,\ldots,n}$ be a set of intervals on the real line representing graph G. We consider the set X of *endpoints* of the intervals:

$$X = \{a_i\}_{i=1,\ldots,n} \cup \{b_i\}_{i=1,\ldots,n} = \{1, \ldots, q\}.$$

Let the *subproblem* $\mathcal{I}(i, j)$ denote the set of all intervals completely contained in the closed interval $[i, j]$. For every pair of values $i \leq j \in X$, let $F(i, j) := \overline{\chi}(G[\mathcal{I}(i, j)], f)$, be the optimum cost of a partition of the subgraph induced by $\mathcal{I}(i, j)$ (by definition of $\overline{\chi}(G, f)$, $F(i, j) = 0$ if $\mathcal{I}(i, j) = \emptyset$). Our Dynamic Programming approach is based on Lemma 5.2 below, which implies that we can separate the problem restricted to $\mathcal{I}(i, j)$ into two subproblems.

Lemma 5.2. *For every $i, j \in X$ there is an optimal partition into cliques of $G[\mathcal{I}(i, j)]$ in which at least one batch induces a maximal clique of $G[\mathcal{I}(i, j)]$.*

Proof. Directly from Proposition 2.2 □

Given $i < z < j \in X$, let $K^z_{i,j}$ be the set of intervals of $\mathcal{I}[i, j]$ containing point z. Notice that $K^z_{i,j}$ is a clique for all $i \leq z \leq j \in X$.

Lemma 5.3. *For arbitrary fixed $i < j$ in X, the following recursion holds:*

$$F(i, j) = \min_{z \in [i,j]} \{f(K^z_{i,j}) + (F(i, z-1) + F(z+1, j))\}. \tag{12}$$

Proof. By Lemma 5.2, there is an optimal partition of $G[\mathcal{I}(i, j)]$ in which a batch is a maximal clique B^*. All maximal cliques of $G[\mathcal{I}(i, j)]$ are browsed while considering the minimum in (12). Hence $B^* = K^{z^*}_{i,j}$ for some z^*. Given such point z^*, every interval in $\mathcal{I}[i, z^* - 1]$ has its terminal endpoint before the initial endpoint of every interval in $\mathcal{I}[z^* + 1, j]$. Hence, the graph $G(\mathcal{I}[i, j] \backslash B^*)$ decomposes into two disconnected subgraphs: $G(\mathcal{I}[x_i, z^* - 1])$ and $G(\mathcal{I}[z^* + 1, j])$. One can therefore solve the problems on these two subgraphs independently. □

The Dynamic Programming algorithm starts from the initial conditions

$$F(i, i) = f(\mathcal{I}[i, i]) \qquad \text{for all } i = 1, \ldots, q.$$

Applying the recursion (12) with increasing subproblem width $x_j - x_i$, it computes an optimal schedule

$$S(x_i, x_j) = \begin{cases} \emptyset & \text{if } \mathcal{I}[i, j] = \emptyset\,; \\ S(i, z^* - 1) \cup B^* \cup S(z^* + 1, j) & \text{otherwise.} \end{cases}$$

The optimum value is $\overline{\chi}(G, f) = F(1, q)$, and $S(1, q)$ is an optimal solution. Since there are $O(q^2) = O(n^2)$ subproblems and $O(q) = O(n)$ candidate values for z in each subproblem, the resulting Dynamic Programming algorithm solves the problem in $O(n^3)$ time. This completes the proof of Theorem 5.1. □

Theorem 5.1 and the associated algorithm can be extended in the following way. A graph $G = (V, E)$ is a *circular arc graph* [13] if there exists a set $\{\phi(v) \mid v \in V\}$ of closed arcs of the unit circle, such that two vertices u and v are adjacent in G if and only if the two corresponding arcs $\phi(u)$ and $\phi(v)$ have nonempty intersection. Call a clique K of G a *Helly clique* if $\cap_{v \in K} \phi(v)$ is nonempty.

Corollary 5.4. *For any circular arc graph G, and any value-polymatroidal function f on $V(G)$ given by a value oracle, we can compute an optimum partition into Helly cliques in time $O(n^3)$.*

Proof. Let X be the set of endpoints of the arcs $\phi(v)$, (as in Theorem 5.1). For $i, j \in X$, let $\mathcal{I}[i, j]$ be the set of arcs contained in the portion of the circle in clockwise order between i and j. Note that after removing any maximal Helly clique, the remaining arcs are contained in some set $\mathcal{I}[i, j]$. Compute all $O(n^2)$ values as in Theorem 5.1. Compute the best maximal Helly clique afterwards. □

On the other hand, we have the following negative result:

Theorem 5.5. [2] *[PCliqW] is NP-hard even if G is an interval graphs and f is a polymatroid cost (even if f is given by a rooted-TSP on a tree).*

Rooted-TSP on trees. Let $T = (W, A)$ be a tree, $l : A \to \mathbb{N}$ and $r \in W$ be the root of T. For $U \subseteq W$, let $A(U)$ be the set of arcs spanning $U + r$ and $f(U) := 2 \sum_{a \in A(U)} l(a)$. The function f is called a rooted-TSP cost since it is the cost of visiting all nodes in $U \subseteq V$, moving along edges of A, starting and finishing the tour from node r (see Fig. 3). Such a cost function can easily be shown to be polymatroidal[1]. Complementing Theorem 5.5, [2] gave a 2-approximation for [PCliqW] when G is an interval graphs and f is rooted-TSP on a tree. This has applications in vehicle routing problems with time windows (where the length $l(a)$ represents a travel cost and we assume that the traveling times are negligible compared to the size of the time windows [9]).

6. Partition into cliques in split graphs

One may wonder if Proposition 2.2 could be applied in more general graphs than interval graphs. A property of interval graphs which is used to prove polynomiality in Theorem 5.1 is that they have a polynomial number of maximal cliques. In this section, we illustrate that this property is not sufficient to ensure polytime solvability of [PCliqW] restricted to value-polymatroidal costs.

A graph $G = (V, E)$ is a *split graph* if V can be partitioned into two sets S and K such that S is a stable set and K is a clique. Notice that split graphs have a polynomial number of maximal cliques (at most $|S| + 1$). However, [max-coloring] and [probabilist coloring] are (strongly) NP-hard in split graphs ([3,8] and [7] respectively). Since the class of split graphs is self-complementary, [PCliqW] is

[1] In fact, several characterizations of the graphs for which rooted TSP costs are polymatroidal for all edge length can be found in [15]. Based on [15], Jost [16] characterized these graphs as the graphs without $K_{2,3}$ minors.

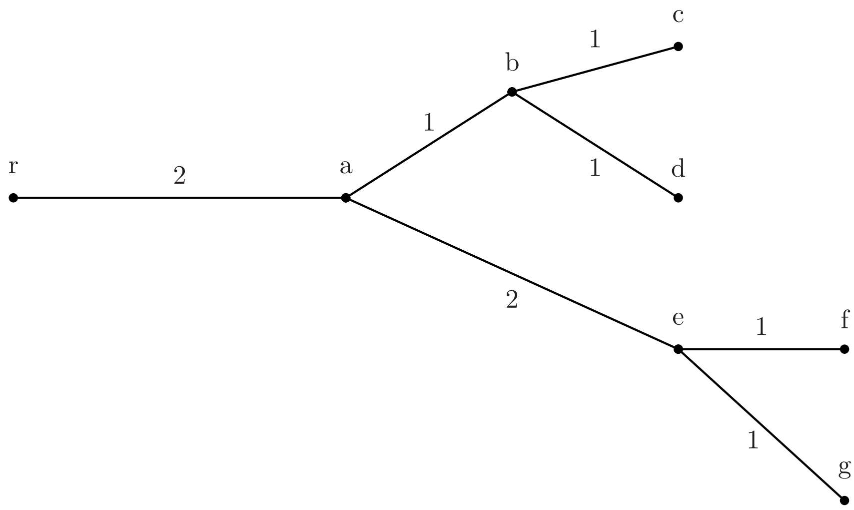

FIGURE 3. A rooted tree with a length function $l : A \to \mathbb{R}$. The cost associated with a subset $U \subseteq V$ is twice the length of the arcs spanning $U + r$. For example $f(\{a\}) = 4$, $f(\{a, b, f\}) = 12$ and $f(\{c, d, e, f\}) = 16$.

also NP-hard if we restrict to maximum or probabilist cost functions. Moreover, Yannakakis and Gavril [18] proved that the maximum q-chromatic subgraph problem is NP-hard for split graph. Unsurprisingly then, Greene-Kleitman's relation doesn't hold for split graphs [5]. However, the "dual problem", that is [PCliqW] with $f(U) := \min\{q, |U|\}$ is trivial. If $q = 1$ this is equivalent to find a partition of G into a minimum number of cliques. If $q \geq 2$, we may assume $\omega(G) = |K|$ (in general, the bipartition (S, K) of a split graph is not unique). Then the partition consisting of all elements of S alone and all vertices of K together in a unique class is optimal. This fact however, does not extend to size-defined cost functions.

Theorem 6.1. *[PCliqW] is strongly NP-hard even if we restrict G to be a split graph and f to be size-defined and value-polymatroidal.*

Proof. We reduce the NP-complete problem [X3C] to [PCliqW].

[X3C] Exact three-set cover
INPUT : A finite set X of size $3m$ and a set T of triples of X.
OUTPUT : Does there exists a partition of X into m elements of T?

Given an instance of [X3C], build the split graph $G = ((T, X), E)$ where $G[T]$ is a stable set and $G[X]$ a clique and $(t, x) \in E$ iff $x \in t$. Let $\psi(0) := 0$, $\psi(1) := \alpha = m+1$ and $\psi(i) := \beta = m+2$ for all $i \geq 2$. We claim that there is a partition of cost not exceeding $m\beta + (|T| - m)\alpha$ if and only if X has a partition into triples of T. A partition into triples yields such a cost. Now, assume that X has no partition into triples. Since T induces a stable set, any partition of $V(G)$ into cliques contains at least $|T|$ classes. Those partitions which consist in exactly $|T|$ cliques, are of cost at least $(m+1)\beta+(|T|-(m+1))\alpha > m\beta+(|T|-m)\alpha$. Those consisting in at least $|T|+1$ cliques are of cost at least $(|T|+1)\alpha > m\beta+(|T|-m)\alpha$. □

7. ILP formulation and min-max formula for [PCliqW]

Seen as a partition problem, [PCliqW] can be formulated as an integer linear program, with variables y in $\mathbb{R}^{\mathcal{C}(G)}$ (where $\mathcal{C}(G)$ is the set of cliques of G):

$$\begin{aligned} &\text{(i)} && \min f^T y; \\ &\text{(ii)} && \sum_{C\ni v} y_C = 1 \text{ for all } v \in V; \\ &\text{(iii)} && y_C \in \{0,1\} \text{ for all } C \in \mathcal{C}(G). \end{aligned} \tag{13}$$

Clearly, if f is non-negative, there is no advantage in taking $y_C > 1$. Therefore, $y_C \in \{0,1\}$ can be replaced by $y_C \geq 0$ and $y_C \in \mathbb{Z}$. Also, if f is non-decreasing, (13) (ii) can be replaced by $\sum_{C\ni v} y_C \geq 1$ (if $y_A = y_B = 1$, $A, B \in \mathcal{C}(G)$ and $A \cap B \neq \emptyset$ then $B \setminus A$ is still a clique of G and we can reset $y_B := 0$ and $y_{B\setminus A} := 1$).

If f is non-negative and non-decreasing, the dual of the linear relaxation of (13) can therefore be written as maximizing $\mathbf{1}^T x$ subject to[2]:

$$\begin{aligned} &\text{(i)} && \sum_{v\in C} x_v \leq f(C) \text{ for all } C \in \mathcal{C}(G); \\ &\text{(ii)} && x_v \geq 0 \text{ for all } v \in V(G). \end{aligned} \tag{14}$$

If G is perfect and $f \equiv 1$, (14) is TDI. Also if G is complete and f is submodular, (14) is box-TDI. So in both cases, (14) yields a min-max formula for [PCliqW]. But there are other famous cases where (14) yields a min-max formula. Greene-Kleitman's theorems can be restated in the following terms: if G is a comparability graph or the complement of such a graph and if f is the rank function of a uniform matroid, system (14) is TDI. Alternatively, Greene-Kleitman's theorems can stated as the box-TDIness of (14) if G is (co)-comparability and $f \equiv 1$ [5]. Note that cliques of the line-graph of a bipartite graph G correpond to subsets of $\delta(v)$ (the set of edges incident with v), for some $v \in V(G)$. Now, a common generalization of the polymatroid intersection theorem, of Dilworth's truncation and of min-max relations for bipartite b-matching can be stated as box-TDIness of (14) if G is the line-graph of a bipartite multigraph and f is submodular. More precisely we have (see Sect. 48.3 of [17] for an idea of the proof and Chapter 60 for extensions),

---

[2]An interpretation of system (14) within the framework of cooperative game theory with cooperation restricted to the cliques of a graph is described in [16].

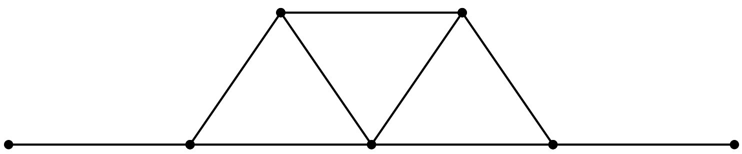

FIGURE 4. Let f be the max-batch cost defined by p. An optimal solution to the linear relaxation of (13) is given by $y_C = 1/2$ if $C \in \{\{v\},\{b,v\},\{a,b,c\},\{a,d,e\},\{c,d\},\{e,w\},\{w\}\}$ and $y_C = 0$ otherwise. The cost of this fractional partition is 13/2. Optimality can be checked using an x maximizing $\mathbf{1}^T x$ subject to (14), for instance $x(a) := 3/2$, $x(c) = x(d) := 1/2$ and $x(b) = x(e) = x(v) = x(w) := 1$.

Theorem 7.1 (submodular bipartite matchings polyhedron [16]). *Let $G = ((A, B), E)$ be a bipartite multi-graph and for all $v \in A \cup B$ let f_v be a submodular function on $\delta(v)$, then the following system is box-TDI*

$$\sum_{e \in F} x_e \leq f_v(F) \text{ for all } v \in A \cup B \text{ and } \emptyset \neq F \subseteq \delta(v). \tag{15}$$

In view of these results, it seems reasonable to expect system (14) to provide other min-max relations for [PCliqW]. However, the linear relaxation of (13) does not always have an integral optimal solution, even if G is an interval graph and f is a value-polymatroidal set function as shown in Figure 4 (other examples for which G is perfect, f is a submodular but the linear relaxation of (13) has no integral optimal solution are provided in [16]).

8. Conclusion and extension

Although we were able to compute an optimum solution for [PCliqW] when G is an interval graph and f is value-polymatroidal, we were unable to complement this result by a min-max formula. This issue could be linked with the following extension: consider the problem of multi-partition into cliques, that is, generalize the ILP (13) by replacing constraints (ii) by $\sum_{C \ni v} y_C = d_v$, where $d_v \in \mathbb{N}$ is the covering demand associated to vertex v. The complexity of this problem is left open and, to the best of our knowledge, is beyond the scope of our dynamic program. A polytime algorithm for this last problem might shed new light on the structure of interval graphs and therefore be useful to solve various problems on interval graphs.

Acknowledgements. This research was supported by the Netherlands Organization for Scientific Research, and by the ADONET network of the European Community, which is a Marie Curie Training Network.

References

[1] N. Alon and A. Orlitsky, Source coding and graph entropies. *IEEE Trans Inform Theory* **42** (1996) 1329–1339.

[2] L. Becchetti, P. Korteweg, A. Marchetti-Spaccamela, M. Skutella, L. Stougie and A. Vitaletti, Latency contrained aggregation in sensor networks. *Workshop of Combinatorial Optimization, Aussois* (2006).

[3] M. Boudhar, Dynamic Scheduling on a Single Batch Processing Machine with Split Compatibility Graphs. *J. Math. Model. Algorithms* **2** (2003) 17–35.

[4] P. Brucker and S. Knust, Complexity results of scheduling problems. `www.mathematik.uni-osnabrueck.de/research/OR/class/`

[5] K. Cameron, A min-max relation for the partial q-colourings of a graph. II: Box perfection. *Discrete Math.* **74** (1989) 15–27.

[6] J. Cardinal, S. Fiorini and G. Joret, Minimum entropy coloring. *ISAAC, Lect. Notes Comput. Sci.* **3827** (2005) 819–828.

[7] F. Della Croce, B. Escoffier, C. Murat and V. Th. Paschos, Probabilistic coloring of bipartite and split graphs. *ICCSA'05, Lect. Notes Comput. Sci.* **3483** (2005) 202–211 (see also *Cahiers du Lamsade* No. 218).

[8] M. Demange, D. de Werra, J. Monnot and V.T. Paschos, Time slot scheduling of compatible jobs. *Cahiers du Lamsade* No. 182, (2001), (accepted in *J. Scheduling*).

[9] E. Desgrippes, Coordination entre la production et la distribution dans une chaîne logistique. *Laboratoire GILCO - Grenoble* (2005).

[10] J. Edmonds and R. Giles, A min-max relation for submodular functions on graphs. *Ann. Discrete Math.* **1** (1977) 185–204.

[11] B. Escoffier, J. Monnot and V. Th. Paschos, Weighted Coloring: Further Complexity and Approximability Results. *ICTCS* (2005) 205–214.

[12] G. Finke, V. Jost, M. Queyranne and A. Sebő, Batch processing with interval graph compatibilities between tasks. *Cahier du Leibniz* No. 108, Laboratoire Leibniz-IMAG, Grenoble (2004) (accepted in *Discrete Appl. Math.*).

[13] M.C. Golumbic, *Algorithmic Graph Theory and Perfect Graphs.* Academic Press (1980).

[14] D.J. Guan and Xuding Zhu, A Coloring Problem for Weighted Graphs. *Inf. Process. Lett.* **61** (1997) 77–81.

[15] Y.T. Herer and M. Penn, Characterizations of natural submodular graphs: A polynomially solvable class of the TSP. *Proc. Am. Math. Soc.* **123** (1995) 673–679.

[16] V. Jost, *Ordonnancement chromatique: Polyèdres, Complexité et Classification.* Ph.D. thesis, Laboratoire Leibniz-IMAG - UJF - Grenoble (2006).

[17] A. Schrijver, *Combinatorial Optimization: Polyhedra and Efficiency.* Springer (2003).

[18] M. Yannakakis and F. Gavril, The maximum k-colorable subgraph problem for chordal graphs. *Inf. Process. Lett.* **24** (1987) 133–137.