In: Proceedings of the Satellite Workshops of ICALP 2000, Proceedings in Informatics 8, Carleton Scientific, 2000, pp. 135–145.

Conversion of Coloring Algorithms into Maximum Weight Independent Set Algorithms

T. ERLEBACH
ETH Zürich, Switzerland

K. JANSEN
Christian-Albrechts-University of Kiel, Germany

Abstract

A very general technique for converting approximation algorithms for the vertex coloring problem in a class of graphs into approximation algorithms for the maximum weight independent set problem (MWIS) in the same class of graphs is presented. The technique consists of solving an LP-relaxation of the MWIS problem with certain clique inequalities, constructing an instance of the vertex coloring problem from the LP solution, applying the coloring algorithm to this instance, and selecting the best resulting color class as the MWIS solution. The approximation ratio obtained is the product of the approximation ratio with which the LP formulation can be solved (usually equal to one) and the approximation ratio of the coloring algorithm with respect to the size of the largest relevant clique. Applying this technique, the best known approximation algorithms are obtained for the maximum weight edge-disjoint paths problem in bidirected trees and in bidirected two-dimensional meshes with row-column routing, and for time-constrained scheduling of weighted packets in the same topologies.

Keywords

Approximation Algorithms, Weighted Independent Set, Edge-Disjoint Paths, Path Coloring, Time-Constrained Packet Scheduling, Linear Programming.

1 Introduction

The maximum independent set problem (MIS) is a fundamental problem in graph theory with many applications. Given a graph $G = (V, E)$, the goal is to compute a subset of the vertices such that no two vertices in the subset are joined by an edge (such a subset is called an independent set) and such that the cardinality of the subset is maximized. In most applications, G represents a conflict graph (an edge joining two vertices u and v indicates that at most one of u and v can be selected) and its vertices represent desirable objects; the goal is to select as many of the desirable objects as possible while not selecting two conflicting objects.

The maximum weight independent set problem (MWIS) is an important generalization of MIS: the vertices of the given graph G are assigned weights, and the goal is to compute an independent set I in G such that the sum of the weights of the vertices in I is maximized.

MIS has long been known to be *NP*-hard for general graphs. More recently, MIS was shown to be very hard to approximate in general: unless $NP = \text{co-}RP$, there can be no polynomial-time approximation algorithm with ratio $n^{1-\varepsilon}$ for any $\varepsilon > 0$ in graphs with n vertices [10]. (Note that an independent set in G is a clique in the complement of G.) For restricted classes of graphs, the situation is different. For example, MIS and MWIS can be solved optimally in polynomial time in perfect graphs [8], in partial k-trees with constant k, in circular-arc graphs, or in the edge-intersection graphs of undirected paths in a tree (if the path representation is given). For other restricted classes of graphs, MIS and MWIS are *NP*-hard, but can be approximated in polynomial time within a constant factor: for example, a $(\frac{5}{3} + \varepsilon)$-approximation for MIS is possible for the edge-intersection graphs of directed paths in bidirected trees [4]. For known approximation results for MIS and MWIS in general graphs and in graphs of bounded degree, please refer to [9] and the references given there.

Bar-Noy et al. obtained an approximation algorithm for throughput maximization on multiple machines, a special case of MWIS, using an LP-relaxation and a rounding procedure based on a coloring subroutine [3]. In this paper, we extend their result and present a general technique for converting vertex coloring algorithms for a class of graphs into MWIS algorithms. The technique consists of solving an LP-relaxation of the MWIS problem (where certain clique inequalities are added to the LP formulation), constructing an instance of the vertex coloring problem from the LP solution, applying the coloring algorithm to this instance, and selecting the best resulting color class as the MWIS solution. The approximation ratio obtained is the product of the approximation ratio with which the LP formulation can be solved (equal to 1 if the LP can be solved optimally in polynomial time) and the approximation ratio of the coloring algorithm with respect to the size of the largest relevant clique. Applying this technique, we obtain new and best known approximation algorithms for the maximum weight edge-disjoint paths problem (MWEDP) in bidirected trees and in bidirected two-dimensional meshes with row-column routing, and substantially improved approximation algorithms for time-constrained scheduling of weighted packets (TCSWP) in the same topologies. The resulting approximation algorithms for the weighted case match the approximation ratios of the best known algorithms for the unweighted case. We expect that the technique will find many further applications (e.g., interval scheduling, frequency assignment, geometric graphs).

1.1 Preliminaries

MIS and MWIS have already been defined at the beginning of the introduction. A ρ-approximation algorithm for MWIS is an algorithm that runs in polynomial time and that always outputs an independent set whose weight is at least a $\frac{1}{\rho}$-fraction of the weight of the optimal solution.

Let $G = (V,E)$ be an undirected graph with vertex weights. The weight of vertex v is denoted by w_v. For $S \subseteq V$, we write $w(S)$ for $\sum_{v \in S} w_v$. The *duplication* of a vertex $v \in V$ is the operation of adding a new vertex v' to G and making v' adjacent to v and to all neighbors of v; the weight of v' is set equal to that of v.

Let $\mathcal{G}$ be a class of graphs that is closed under duplication of vertices and under taking of (induced) subgraphs. For every graph $G \in \mathcal{G}$, let $\mathcal{C}(G)$ be a set of cliques in G (e.g., the set of all maximal cliques or, in circular-arc graphs, the set of all cliques of arcs overlapping one point of the circle). We call $\mathcal{C}(G)$ the *relevant cliques* of graph G. By $\omega_{\mathcal{C}}(G)$ we denote the cardinality $\max_{C \in \mathcal{C}(G)} |C|$ of the largest relevant clique of G. In many applications the graph G will be given implicitly in another representation (e.g., G is taken to be the conflict graph of given paths in a network or of given intervals on the real line), and we allow the relevant cliques $\mathcal{C}(G)$ to be defined in terms of the given representation of G.

If a graph $G' = (V',E')$ is obtained from $G = (V,E)$ by deleting vertices and duplicating every remaining vertex an arbitrary number of times, we require sometimes that the set $\mathcal{C}(G')$ of relevant cliques in G' must contain exactly those cliques C' that can be obtained from a clique $C \in \mathcal{C}(G)$ by deleting and duplicating vertices in the same way as for obtaining G' from G. This requirement, which we call the *clique propagation* property, may seem restrictive, but it is satisfied automatically by natural definitions of $\mathcal{C}(G)$ in many applications (see Sections 3 and 4).

Given a graph $G = (V,E)$, a *coloring* of G is an assignment of colors to vertices such that adjacent vertices receive different colors. The minimum number of colors required in a coloring is the chromatic number $\chi(G)$. Every color class (set of vertices assigned the same color) is an independent set in G.

For a given graph $G = (V,E)$ with vertex weights $0 \leq c_v \leq 1$, $v \in V$, a *weighted fractional coloring* of G is a solution to the following linear program LP_w, where $\mathcal{I}$ is the set of all independent sets in G and a variable x_I indicates the "length" of the independent set I:

$$\begin{array}{lll} (LP_w) & \min & \sum_{I \in \mathcal{I}} x_I \\ & \text{s.t.} & \sum_{I \in \mathcal{I}: v \in I} x_I \geq c_v \quad \forall v \in V, \\ & & \qquad 1 \geq x_I \geq 0 \quad \forall I \in \mathcal{I}. \end{array}$$

The goal is to compute a partition into independent sets I with fractions $x_I \in [0,1]$ such that each vertex v is covered at least by c_v and such that the partition has minimum total length $\sum_{I \in \mathcal{I}} x_I$. This minimum total length is the weighted fractional chromatic number $\chi_{f,w}(G,c)$ of G with weight vector c. Define $\omega_{\mathcal{C}}(G,c) =$

$\max_{C \in \mathcal{C}(G)} \sum_{v \in C} c_v$. It is clear that $\omega_{\mathcal{C}}(G,c)$ is a lower bound for $\chi_{f,w}(G,c)$. Any feasible solution to LP_w is a weighted fractional coloring, and we refer to the corresponding objective value of LP_w as the *total length* of the coloring.

Note that the size of LP_w can be exponential in the size of the graph; an algorithm that computes a solution to LP_w in polynomial time (polynomial in the size of the graph and in the size of the representation of the vertex weights c_v) cannot construct LP_w explicitly.

If $c_v = 1$ for all $v \in V$, weighted fractional colorings are simply called *fractional colorings*. The minimum total length of a fractional coloring is the fractional chromatic number $\chi_f(G)$. Note that $\omega_{\mathcal{C}}(G) \leq \chi_f(G) \leq \chi(G)$.

We will use the following alternative interpretation of weighted fractional colorings: A weighted fractional coloring using independent sets of total length K can be seen as an assignment of open subintervals of $(0,K)$ to vertices of G such that every vertex v is assigned disjoint subintervals of total length at least c_v and such that adjacent vertices are assigned disjoint subintervals.

Now we define a linear program to compute a maximum weight fractional independent set subject to clique constraints for all cliques in $\mathcal{C}(G)$, using a variable x_v for every vertex $v \in V$ to indicate the fraction of vertex v that should be selected:

$$\begin{array}{lll} (LP_I) & \max & \sum_{v\in V} w_v x_v \\ & \text{s.t.} & \sum_{v \in C} x_v \leq 1 \quad \forall C \in \mathcal{C}(G), \\ & & 0 \leq x_v \leq 1 \quad \forall v \in V. \end{array}$$

If $\mathcal{C}(G)$ is the set of all maximal cliques in G, the feasible solutions to LP_I form the so-called *fractional node-packing polytope $QSTAB(G)$* [8].

For the following definitions, we assume that this linear program can be solved (for the graph class $\mathcal{G}$ and relevant cliques $\mathcal{C}(G)$ for $G \in \mathcal{G}$) either optimally in polynomial time or approximatively with ratio $\beta \geq 1$. In the optimum case, we have $\beta = 1$. We say that

1. $(\mathcal{G}, \mathcal{C})$ is an (α,β)*-tractable* class of graphs if the clique propagation property holds and if there is a polynomial-time algorithm to solve the coloring problem using at most $\alpha\omega_{\mathcal{C}}(G)$ colors for every graph $G \in \mathcal{G}$,

2. $(\mathcal{G}, \mathcal{C})$ is (α,β)*-fractional-tractable* if the clique propagation property holds and if a fractional coloring of total length at most $\alpha\omega_{\mathcal{C}}(G)$ can be computed in polynomial time for every graph $G \in \mathcal{G}$, and

3. $(\mathcal{G}, \mathcal{C})$ is (α,β)*-weighted-fractional-tractable* if a weighted fractional coloring of total length at most $\alpha\omega_{\mathcal{C}}(G,c)$ can be computed in polynomial time for all graphs $G \in \mathcal{G}$ with arbitrary weight vectors c satisfying $0 \leq c_v \leq 1$ for all $v \in V$.

Note that an (α,β)-tractable class of graphs is also (α,β)-fractional-tractable.

1.2 Related Work on Coloring and Fractional Coloring

Grötschel et al. [7] have shown that the weighted fractional coloring problem is *NP*-hard for general graphs, but can be solved in polynomial time on perfect graphs. Specifically, they have proved the following result: For any graph class $\mathcal{G}$, if the problem of computing $\alpha(G)$, the size of the largest independent set in G, for graphs $G \in \mathcal{G}$ is *NP*-hard, then the problem of determining the weighted fractional chromatic number is also *NP*-hard. This gives a negative result for the weighted fractional coloring problem even for planar cubic graphs. Furthermore, if MWIS is polynomial-time solvable for graphs in $\mathcal{G}$, then the weighted fractional coloring problem for $\mathcal{G}$ can also be solved in polynomial time.

Lund and Yannakakis [12] proved that there is a $\delta > 0$ such that there does not exist a polynomial-time approximation algorithm for fractional coloring that achieves ratio $|V|^{\delta}$, unless $P = NP$.

Feige and Kilian [6] proved that the chromatic number cannot be approximated within $|V|^{1-\varepsilon}$ for any $\varepsilon > 0$, unless $NP \subseteq$ ZPP. This also holds for the fractional coloring problem, due to the logarithmic relationship (see also [11]) of the fractional chromatic number and the chromatic number (i.e. $\chi_f(G) \leq \chi(G) \leq \chi_f(G)(1 + \ln \alpha(G))$).

2 The Conversion Technique

Using the definitions from Section 1.1, we can state our main results as follows.

Theorem 1 *For every class of graphs that is (α,β)-tractable or (α,β)-fractional-tractable, there is an approximation algorithm for MWIS that achieves approximation ratio $\alpha\beta + \varepsilon$, where ε is an arbitrarily small positive constant.*

Theorem 2 *For every (α,β)-weighted-fractional-tractable class of graphs, there is an approximation algorithm for MWIS that achieves approximation ratio $\alpha\beta$.*

Let $(\mathcal{G}, \mathcal{C})$ be an (α,β)-fractional-tractable class of graphs. Thus we have an approximation algorithm A that computes, for any graph $G \in \mathcal{G}$, a fractional coloring of total length at most $\alpha\omega_{\mathcal{C}}(G)$.

Let $G = (V,E)$ be a graph from $\mathcal{G}$ with positive weights w_v for vertices $v \in V$. Denote the optimum value of LP_I by z^*. By our assumptions, we have an algorithm B that computes a solution x^B of value $z^B \geq \frac{1}{\beta} z^*$ in polynomial time. Let $\varepsilon > 0$ be an arbitrary positive constant. We define $\delta = \frac{\varepsilon}{\alpha\beta+\varepsilon}$ and $N = \lceil 4|V|/\delta^2 \rceil$. Let $\bar{x}_v = (1-\delta/2)x_v^B + \frac{\delta}{2|V|}$. It is easy to see that $\bar{x}$ is a feasible solution of LP_I with objective value $\sum_{v \in V} w_v \bar{x}_v \geq (1-\delta/2)\frac{1}{\beta}z^*$. To get a solution $\hat{x}$, we round the $\bar{x}_v$ down to the largest possible multiple of $\frac{1}{N}$ (i.e. $\hat{x}_v = \lfloor \bar{x}_v N \rfloor \cdot \frac{1}{N}$). We obtain that

$$\hat{x}_v \geq \bar{x}_v - \frac{1}{N} \geq \bar{x}_v - \frac{\delta^2}{4|V|} = \bar{x}_v - \frac{\delta}{2|V|} \cdot \frac{1}{2}\delta \geq \bar{x}_v(1-\delta/2)$$

using $\bar{x}_v \geq \frac{\delta}{2|V|}$. Since $\hat{x}$ is generated by rounding the $\bar{x}_v$ down to a multiple of $\frac{1}{N}$, feasibility of $\bar{x}$ for LP_I implies also feasibility of $\hat{x}$ for LP_I. The objective value $\hat{z}$ of $\hat{x}$ is given by $\sum_{v\in V} w_v\hat{x}_v \geq (1-\delta/2)\sum_{v\in V} w_v\bar{x}_v \geq (1-\delta/2)^2\frac{1}{\beta}z^* \geq (1-\delta)\frac{1}{\beta}z^*$.

So we have a rounded solution $\hat{x}$ to LP_I such that every $\hat{x}_v$ is of the form $\frac{k}{N}$ for some $k \in \{0,1\ldots,N\}$ and such that the objective value $\hat{z} \geq (1-\delta)\frac{1}{\beta}z^*$.

Next, we construct a graph $\hat{G}$ by using the rounded solution $\hat{x}$ of LP_I. The set $\hat{V}$ of vertices of $\hat{G}$ contains, for every vertex $v \in V$, exactly k copies of v provided that $\hat{x}_v = \frac{k}{N}$. To obtain $\hat{G}$ from G, we replace vertex v by a clique A_v with k vertices such that each vertex in A_v has the same neighborhood (outside A_v) as v; in other words, vertex v is duplicated $k-1$ times if $k \geq 1$ and is deleted if $k=0$.

By the clique propagation property, each relevant clique $\hat{C} \in \mathcal{C}(\hat{G})$ is generated from a clique $C \in \mathcal{C}(G)$, and this implies $|\hat{C}| = \sum_{v\in C}|A_v| = \sum_{v\in C}\hat{x}_v \cdot N \leq N$. Therefore, $\omega_{\mathcal{C}}(\hat{G}) \leq N$.

Finally, we use algorithm A to compute a fractional coloring of $\hat{G}$ with total length at most $\alpha\omega_{\mathcal{C}}(\hat{G}) \leq \alpha N$ (and with only a polynomial number of independent sets $U_1,\ldots,U_y$). Since every vertex is covered by at least one, we have $\sum_{i=1}^{y} x_{U_i} w(U_i) \geq \hat{z}N$. Furthermore, the length $\sum_{i=1}^{y} x_{U_i} \leq \alpha N$. Suppose that all independent sets have weight $w(U_i) < \frac{\hat{z}}{\alpha}$. Then, the sum $\sum_{i=1}^{y} x_{U_i} w(U_i) < \sum_{i=1}^{y}\frac{x_{U_i}\hat{z}}{\alpha} = \frac{\hat{z}}{\alpha}\sum_{i=1}^{y} x_{U_i} \leq \hat{z}N$. This gives a contradiction. Therefore, there is at least one independent set U_i with $w(U_i) \geq \frac{\hat{z}}{\alpha} \geq \frac{1}{\alpha\beta+\varepsilon}z^*$. We obtain an $(\alpha\beta+\varepsilon)$-approximation algorithm for MWIS. At this point, we have proved Theorem 1. (Recall that an (α,β)-tractable class of graphs is also (α,β)-fractional-tractable.)

Now assume that $(\mathcal{G},\mathcal{C})$ is an (α,β)-weighted-fractional-tractable class of graphs. We have an algorithm $A_{f,w}$ to compute weighted fractional colorings with total length at most $\alpha\omega_{\mathcal{C}}(G,c)$ for any vertex weight vector c. After obtaining solution x^B to LP_I, we set $c_v = x_v^B$ for all $v \in V$ and use $A_{f,w}$ to compute a weighted fractional coloring of (G,x^B) with total length at most $\alpha\omega_{\mathcal{C}}(G,x^B)$ and with a polynomial number of independent sets $U_1,\ldots,U_y$. This solution of LP_w satisfies $\sum_{i=1}^{y} w(U_i)x_{U_i} \geq z^B$. We can bound $\omega_{\mathcal{C}}(G,x^B) = \max_{C\in\mathcal{C}(G)}\sum_{v\in C} x_v^B \leq 1$ since x^B is a feasible solution of LP_I. Again, supposing that all independent sets in the solution of algorithm $A_{f,w}$ have weight less than $\frac{z^B}{\alpha}$ gives a contradiction. Therefore, one of the independent sets has weight $\geq \frac{z^B}{\alpha} \geq \frac{z^*}{\alpha\beta}$. This gives us an $\alpha\beta$-approximation algorithm for MWIS, thus proving Theorem 2.

3 Weighted Edge-Disjoint Paths Problems

A directed graph $G=(V,E)$ is *bidirected* if $(u,v) \in E$ implies $(v,u) \in E$. A bidirected tree is the graph obtained from an undirected tree by replacing each edge by two directed edges with opposite directions. A bidirected mesh is obtained in the same way from an undirected mesh. Here (and in the next section) we assume that all paths in meshes are row-column paths (dimension-order paths), i.e., paths that consist of a horizontal segment in the row of the starting node followed by a vertical segment in the column of the end node of the path. Row-column routing is often used in practice because it is deadlock-free and can be implemented with simple hardware.

An instance of the maximum weight edge-disjoint paths problem (MWEDP) consists of a bidirected graph $G = (V,E)$ and a set P of directed paths in G, where each path $p \in P$ has a weight $w_p > 0$. A feasible solution is a subset of the given paths (the *accepted* paths) such that the paths in that subset are edge-disjoint. The goal is to maximize the total weight of the accepted paths. Edge-disjoint paths problems of this type lie at the heart of resource allocation problems in communication networks that support bandwidth reservation.

Given a set P of directed paths in a bidirected graph $G = (V,E)$, the *load* $L(e)$ of an edge $e \in E$ is the number of paths in P that go through edge e. The *maximum load* L is then defined as $\max_{e\in E} L(e)$. A *coloring* of the paths in P is an assignment of colors such that paths receive different colors if they share a directed edge. The *path coloring* problem is the problem of coloring a given set of paths with the minimum number of colors.

The unweighted version of MWEDP (where all paths have the same weight), called MEDP, has been studied for tree networks in [4]. It was shown that MEDP can be solved optimally in polynomial time for bidirected trees of constant degree and is MAX SNP-hard for bidirected trees of arbitrary degree. For every fixed $\varepsilon > 0$, a $(\frac{5}{3} + \varepsilon)$-approximation algorithm was presented. The best known approximation algorithm for path coloring in bidirected trees uses at most $5L/3$ colors, where L is the maximum load [5].

3.1 Approximating MWEDP in Trees and Meshes

Let $\mathcal{G}$ be the class of all conflict graphs obtained from sets of directed paths in bidirected trees. For a given set of directed paths in a bidirected tree, we let G be the conflict graph (edge-intersection graph) of the paths and $\mathcal{C}(G)$ be the set of all cliques in G corresponding to paths using the same directed edge of the tree. Note that the clique propagation property is satisfied. Obviously, $\omega_{\mathcal{C}}(G) = L$, where L is the maximum load. LP_I has polynomial size and can be solved optimally in polynomial time ($\beta = 1$), and the coloring algorithm from [5] uses at most $5L/3$ colors ($\alpha = \frac{5}{3}$). Therefore, $(\mathcal{G},\mathcal{C})$ is a $(\frac{5}{3},1)$-tractable class of graphs, and by Theorem 1 we get a $(\frac{5}{3} + \varepsilon)$-approximation algorithm for MWEDP in bidirected trees.

We have the following complexity result for meshes with row-column routing.

Theorem 3 *MEDP is NP-hard for meshes with row-column routing. Path coloring is NP-hard for meshes with row-column routing even if the number of rows is three.*

The proof of Theorem 3 is omitted due to space limitations.

Let $\mathcal{G}$ be the class of all conflict graphs obtained from sets of directed row-column paths in bidirected meshes. For a given set of directed row-column paths in a bidirected mesh, let G and $\mathcal{C}(G)$ be defined as above. LP_I can again be solved optimally in polynomial time ($\beta = 1$), giving a solution vector x^B. The following procedure computes a weighted fractional coloring of (G, x^B) using independent sets of total length at most $4\omega_{\mathcal{C}}(G, x^B)$. Partition the given paths into the sets right-up, right-down, left-up, and left-down. Vertical and horizontal paths can be put in any of the two possible sets. The paths in one of the four sets can be colored using independent sets of total length at most $2\omega_{\mathcal{C}}(G, x^B)$, as will be shown below. Then we can use the same colors for the sets right-up and left-down, and for the sets right-down and left-up. To color the paths in one set, say right-up, consider the nodes of the mesh starting from the bottom right corner, processing the nodes in one row from right to left and processing the rows from bottom to top. At each node, consider the uncolored paths that turn (or start or end) at that node in arbitrary order, and assign each of them free subintervals of free colors greedily. Let p denote the currently processed path. In order to color p, we must assign disjoint subintervals of $(0, 2\omega_{\mathcal{C}}(G, x^B))$ of total length x^B_p to p. As previously colored paths that intersect the current path must intersect its last horizontal edge or its first vertical edge, at most $2(\omega_{\mathcal{C}}(G, x^B) - x^B_p)$ of the available colors (i.e., of the interval $(0, 2\omega_{\mathcal{C}}(G, x^B)))$ are blocked, and free intervals of the required total length can always be found.

Note that the coloring procedure can be implemented in polynomial time. At any time during the coloring procedure, there exist a certain number of interval endpoints that have been used for specifying subintervals so far. When path p is colored, the required subintervals can be chosen such that at most one new interval endpoint is introduced; hence, there will be only $O(|P|)$ interval endpoints, and they can be processed in polynomial time.

We have proved that $(\mathcal{G}, \mathcal{C})$ is $(4,1)$-weighted-fractional-tractable, so from Theorem 2 we get a 4-approximation for MWEDP in meshes with row-column routing.

Theorem 4 *There are approximation algorithms with approximation ratio* $\frac{5}{3} + \varepsilon$ *(for every fixed* $\varepsilon > 0$*) for MWEDP in bidirected trees and with approximation ratio* 4 *for MWEDP in bidirected meshes with row-column routing.*

4 Time-Constrained Packet Scheduling

For time-constrained scheduling of weighted packets (TCSWP) in a bidirected network, every packet p is specified by a directed path P_p, a release time r_p, a deadline d_p, and a weight $w_p > 0$. The number of links in P_p is denoted by $|P_p|$. We consider the bufferless case, where a packet must continue to travel one link in every time step after it has left the sender until it reaches the receiver. Transmitting a packet across one link takes exactly one time step, and every link can transmit at most one packet in every time step. A node can send and receive several packets simultaneously (multi-port model). An instance of TCSWP consists of a bidirected graph and a set of packets. Let n denote the number of packets. A feasible solution selects a subset of the given packets (the *accepted* packets) and specifies a starting time $s_p \geq r_p$ for every accepted packet, where every accepted packet must reach its destination by its deadline (i.e., $s_p + |P_p| \leq d_p$) and no two packets can use the same link in the same time step. The goal is to maximize the sum of the weights of the accepted packets.

TCSWP was studied by Adler et al. for unweighted packets in linear networks (chains) [2]. They proved the problem *NP*-hard in the bufferless case and presented a 2-approximation algorithm. For packets with arbitrary weights, an 18-approximation algorithm for bidirected trees and an $O(1)$-approximation algorithm for bidirected meshes with row-column routing was presented in [1]. For unweighted packets, they gave a 3-approximation for bidirected trees.

4.1 Approximating TCSWP in Trees and Meshes

Let t be a possible start time for packet p, i.e., $r_p \leq t \leq d_p - |P_p|$. The *$t$-realization* of packet p is a packet that traverses the i-th link of P_p in time step $t + i - 1$, $1 \leq i \leq |P_p|$. The t_1-realization of packet p_1 and the t_2-realization of packet p_2 are in conflict either if $p_1 = p_2$ or if there is a directed edge e that is used in the same time step by both realizations. Now, TCSWP can be viewed as the problem of selecting a maximum weight subset of all possible realizations of all given packets such that no two selected realizations are in conflict.

Given a set Q of realizations of packets, a coloring of Q is an assignment of colors to elements of Q such that realizations that are in conflict receive different colors.

For a given instance of TCSWP, the number of possible packet realizations may be exponential in the size of the input. However, this is not a problem because of the following arguments: In trees and in meshes with row-column routing, a packet p with $d_p - |P_p| - r_p > n$ can always be scheduled: after all $n-1$ other packets are scheduled, there still exists at least one start time s_p for packet p such that the s_p-realization of p is not in conflict with any of the other scheduled packets. Therefore, all packets p with $d_p - |P_p| - r_p > n$ can be removed from the instance and treated separately. Then $d_p - r_p$ is polynomial in the size of the input for all remaining packets, and only a polynomial number of packet realizations must be considered.

First, let $\mathcal{G}$ be the class of all graphs obtained as conflict graphs of packet realizations in bidirected tree networks. For a graph $G \in \mathcal{G}$, let $\mathcal{C}(G)$ be the set of all cliques containing packet realizations using the same edge of the network in the same time step and all cliques containing packet realizations belonging to the same original packet. $(\mathcal{G},\mathcal{C})$ is a $(3,1)$-weighted-fractional-tractable class of graphs. For a given graph $G \in \mathcal{G}$ with vertex weights w_v, LP_I has polynomial size and can be solved optimally in polynomial time, giving a solution vector x^B. Then the following procedure colors (G,x^B) using independent sets of total length at most $3\omega_{\mathcal{C}}(G,x^B)$. Process the nodes of the tree in a pre-order traversal starting at the root. At each node v, consider the packet realizations whose paths touch (start, end, or go through) the node v but no node that is closer to the root. Assign each such realization r the smallest available free color intervals. It is easy to see that, if a previously colored packet realization uses an edge in the same time step as r, it must in fact use an edge incident to v in the same time step as r. Hence, at most $3(\omega_{\mathcal{C}}(G,x^B)-x_r^B)$ of the available colors are used by previously colored realizations that are in conflict with r, namely at most $2(\omega_{\mathcal{C}}(G,x^B)-x_r^B)$ by realizations using an edge incident to v in the same time step as r and at most $\omega_{\mathcal{C}}(G,x^B)-x_r^B$ by realizations belonging to the same packet as r. Therefore, $3\omega_{\mathcal{C}}(G,x^B)$ colors suffice to complete the weighted fractional coloring. By Theorem 2 we obtain a 3-approximation algorithm for TCSWP in bidirected trees.

Now let $\mathcal{G}$ be the class of all graphs obtained as conflict graphs of packet realizations in bidirected mesh networks. For a graph $G \in \mathcal{G}$, let $\mathcal{C}(G)$ be defined as above. Using similar arguments as above and as for MWEDP in meshes (Section 3), we obtain that $(\mathcal{G},\mathcal{C})$ is a $(6,1)$-weighted-fractional-tractable class of graphs. By Theorem 2 we get a 6-approximation algorithm.

Theorem 5 *There are approximation algorithms with approximation ratio* 3 *for TCSWP in bidirected trees and with approximation ratio* 6 *for TCSWP in bidirected meshes with row-column routing.*

We note that our approximation algorithms for TCSWP work also if every packet does not specify a release time and a deadline, but an arbitrary (possibly non-contiguous) set of possible start times.

Acknowledgments

The authors are grateful to Baruch Schieber and Seffi Naor for bringing the work in [3] to their attention and for helpful discussions.

References

[1] ADLER, M., KHANNA, S., RAJARAMAN, R., AND ROSÉN, A. Time-Constrained Scheduling of Weighted Packets on Trees and Meshes. In *Proceedings of the 11th Annual ACM Symposium on Parallel Algorithms and Architectures* (1999), pp. 1–12.

[2] ADLER, M., ROSENBERG, A. L., SITARAMAN, R. K., AND UNGER, W. Scheduling Time-Constrained Communication in Linear Networks. In *Proceedings of the 10th Annual ACM Symposium on Parallel Algorithms and Architectures* (1998), pp. 269–278.

[3] BAR-NOY, A., GUHA, S., NAOR, J. S., AND SCHIEBER, B. Approximating the throughput of multiple machines under real-time scheduling. In *Proceedings of the 31st Annual ACM Symposium on Theory of Computing* (1999), pp. 622–631.

[4] ERLEBACH, T., AND JANSEN, K. Maximizing the number of connections in optical tree networks. In *Proceedings of the 9th Annual International Symposium on Algorithms and Computation* (1998), LNCS 1533, pp. 179–188.

[5] ERLEBACH, T., JANSEN, K., KAKLAMANIS, C., MIHAIL, M., AND PERSIANO, P. Optimal wavelength routing on directed fiber trees. *Theoretical Computer Science 221* (1999), 119–137. Special issue of ICALP'97.

[6] FEIGE, U., AND KILIAN, J. Zero knowledge and the chromatic number. *Journal of Computer and System Sciences 57*, 2 (1998), 187–199.

[7] GRÖTSCHEL, M., LOVÁSZ, L., AND SCHRIJVER, A. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica 1* (1981), 169–197. Corrigendum in Combinatorica 4, 291-295.

[8] GRÖTSCHEL, M., LOVÁSZ, L., AND SCHRIJVER, A. *Geometric Algorithms and Combinatorial Optimization*. Springer-Verlag, Berlin, 1988.

[9] HALLDÓRSSON, M. M. Approximations of weighted independent set and hereditary subset problems. In *Proceedings of the 5th International Computing and Combinatorics Conference* (1999), LNCS 1627, pp. 261–270.

[10] HÅSTAD, J. Clique is hard to approximate within $n^{1-\varepsilon}$. In *Proceedings of the 37th Annual Symposium on Foundations of Computer Science* (1996), pp. 627–636.

[11] LOVÁSZ, L. On the ratio of the optimal integral and fractional covers. *Discrete Mathematics 13* (1975), 383–390.

[12] LUND, C., AND YANNAKAKIS, M. On the hardness of approximating minimization problems. *Journal of the ACM 41*, 5 (1994), 960–981.

Thomas Erlebach is with the Computer Engineering and Networks Laboratory (TIK) at ETH Zürich, CH-8092 Zürich, Switzerland. E-mail: erlebach@tik.ee.ethz.ch

Klaus Jansen is with the Institute of Computer Science and Applied Mathematics in the Technical Faculty of Christian-Albrechts-University of Kiel, Olshausenstr. 40, D-24098 Kiel, Germany. E-mail: kj@informatik.uni-kiel.de