

Bandwidth Constrained Multi-interface Networks

Gianlorenzo d'Angelo, Gabriele Di Stefano, Alfredo Navarra

► **To cite this version:**

Gianlorenzo d'Angelo, Gabriele Di Stefano, Alfredo Navarra. Bandwidth Constrained Multi-interface Networks. 37th Conference on Current Trends in Theory and Practice of Computer Science, Jan 2011, Nový Smokovec, Slovakia. pp.202-213, 10.1007/978-3-642-18381-2_17 . hal-00644104

HAL Id: hal-00644104

<https://hal.inria.fr/hal-00644104>

Submitted on 23 Nov 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Bandwidth constrained multi-interface networks

Gianlorenzo D'Angelo¹, Gabriele Di Stefano¹, and Alfredo Navarra²

¹ Dipartimento di Ingegneria Elettrica e dell'Informazione, Università degli Studi dell'Aquila, Italy. gianlorenzo.dangelo@univaq.it gabriele.distefano@univaq.it

² Dipartimento di Matematica e Informatica, Università degli Studi di Perugia, Italy. navarra@dmi.unipg.it

Abstract. In heterogeneous networks, devices can communicate by means of multiple wired or wireless interfaces. By switching among interfaces or by combining the available interfaces, each device might establish several connections. A connection is established when the devices at its endpoints share at least one active interface. Each interface is assumed to require an activation cost, and provides a communication bandwidth. In this paper, we consider the problem of activating the cheapest set of interfaces among a network $G = (V, E)$ in order to guarantee a minimum bandwidth B of communication between two specified nodes. Nodes V represent the devices, edges E represent the connections that can be established. In practical cases, a bounded number k of different interfaces among all the devices can be considered. Despite this assumption, the problem turns out to be *NP*-hard even for small values of k and Δ , where Δ is the maximum degree of the network. In particular, the problem is *NP*-hard for any fixed $k \geq 2$ and $\Delta \geq 3$, while it is polynomially solvable when $k = 1$, or $\Delta \leq 2$ and $k = O(1)$. Moreover, we show that the problem is not approximable within $\eta \log B$ or $\Omega(\log \log |V|)$ for any fixed $k \geq 3$, $\Delta \geq 3$, and for a certain constant η , unless $P = NP$. We then provide an approximation algorithm with ratio guarantee of b_{max} , where b_{max} is the maximum communication bandwidth allowed among all the available interfaces. Finally, we focus on particular cases by providing complexity results and polynomial algorithms for $\Delta \leq 2$.

1 Introduction

The interest in wireless networks has rapidly grown during the last decades. Their success is certainly due to the wide range of applications for which such networks are designed. A very important issue is constituted by the heterogeneity of the devices which might interact in order to exchange data. Wireless networks are, in fact, composed of devices with different characteristics like computational power, energy consumption, radio interfaces, supported communication protocols, and so forth. In this paper, we are mainly interested in devices equipped with multiple interfaces (like Bluetooth, WiFi, GPRS, etc.). A connection between two or more devices might be accomplished by means of different communication networks according to connectivity and quality of service requirements. The selection of the most suitable interface for a specific connection might depend on various factors. Such factors include: its availability in specific devices,

the required communication bandwidth, the cost (in terms of energy consumption) of maintaining an active interface, the available neighbors, and so forth. While managing such connections, a lot of effort must be devoted to energy consumption issues. Devices are, in fact, usually battery powered and the network survivability might depend on their persistence in the network.

We study communication problems in wireless networks supporting multiple interfaces. In the considered model, the input network is described by a graph $G = (V, E)$, where V represents the set of wireless devices and E is the set of possible connections according to proximity of devices and the available interfaces that they may share. Each $v \in V$ is associated with a set of available interfaces $W(v)$. The set of all the possible interfaces available in the network is then determined by $\bigcup_{v \in V} W(v)$; we denote the cardinality of this set by k . We say that a connection is satisfied (or covered) when the endpoints of the corresponding edge share at least one active interface. If an interface x is activated at some node u , then u consumes some energy $c(x)$ for maintaining x as active, and it provides a maximum communication bandwidth $b(x)$ with all its neighbors which share interface x . In this setting, we study the problem of establishing a communication path between two selected nodes $s, t \in V$ of minimum cost in terms of energy consumption, while guaranteeing a minimum communication bandwidth B . In other words, we look for the minimum cost set of active interfaces among the input graph G , in such a way that s is guaranteed to exchange data with t at least with some bandwidth B . This implies that between s and t not necessarily a path of covered edges must be established but a more complex graph might be required according to the topology and to the available interfaces.

Related work. Multi-interface wireless networks have recently been studied in a variety of contexts, usually focusing on the benefits of multiple radio devices of each node. Many basic problems of standard wireless network optimization can be reconsidered in such a setting [3], in particular, focusing on issues related to routing [7] and network connectivity [5, 8]. The study of combinatorial problems on multi-interface wireless networks has originated from [4]. That paper, as well as [13] investigate the so called *Coverage* problem, where the goal is the activation of the minimum cost set of interfaces in such a way that all the edges of G are covered. *Connectivity* issues have been addressed in [2, 6, 14]. The goal becomes to activate the minimum cost set of interfaces in G in order to guarantee a path of communication between every pair of nodes. In [14], the attention has been devoted to the so called *Cheapest path* problem. This corresponds to the well-known shortest path problem but in the context of multi-interface networks. A natural continuation on investigating such kind of networks is certainly to consider also quality of service constraints in the problem. To the best of our knowledge, bandwidth issues have never been treated before in this context.

Our results. In this paper, we are interested in establishing the cheapest way of communication between two given nodes while guaranteeing a minimum bandwidth of communication. The resulting problem, called *Bandwidth Constraints in Multi-Interface Networks (BCMI)* is similar to the better known *Minimum Edge Cost Flow* [9]. The main difference resides in the fact that we do not con-

Δ	k	Complexity
$\Delta = 1$	Bounded	Optimally solvable in $O(1)$ time
	Unbounded	NP-hard (equiv. MinKnapsack), $(1 + \epsilon)$ -apx in $O(\frac{k^2}{\epsilon})$
$\Delta = 2$	Bounded	Optimally solvable in $O(V)$
	Unbounded	NP-hard; $(2 + \epsilon)$ -apx in $O(V ^{\frac{k^2}{\epsilon}})$ for paths
Fixed $\Delta \geq 3$	Fixed $k \geq 2$	NP-hard (from $X3C$)
	Fixed $k \geq 3$	Not apx within $\eta \log B$, or within $\Omega(\log \log V)$
Any	$k = 1$	Opt. solvable in $O(V + E)$ (equiv. Shortest Path)
	Any	b_{\max} -apx (optimal for constant bandwidth)

Table 1. Complexity results achieved in this paper for *BCMI*

sider costs and capacities for the edges of the network but we have to cope with interfaces at the nodes that require some costs and can manage some maximum bandwidths. In the special case where each connection can be established by means of a different interface, the two problems coincide. Hence, it is not surprising that *BCMI* turns out to be *NP*-hard when the number k of interfaces is unbounded. However, in practical cases it is more realistic to consider a bounded number of interfaces. Despite the expectations, we show that the problem is *NP*-hard even when k is a fixed small number. In detail, we prove that the problem is *NP*-hard for any fixed $k \geq 2$ and $\Delta \geq 3$, where Δ is the maximum degree of the network, while it is polynomially solvable when $k = 1$, or $\Delta \leq 2$ and $k = O(1)$. Moreover, we show that the problem is not approximable within $\eta \log B$ or $\Omega(\log \log |V|)$ for any fixed $k \geq 3$, $\Delta \geq 3$, and for a certain constant η , unless $P = NP$. We then provide an approximation algorithm with ratio guarantee of b_{max} , where b_{max} is the maximum communication bandwidth allowed among all the available interfaces. This algorithm optimally solves the problem in the case that the bandwidth is constant for all the interfaces. Finally, we focus on particular cases by providing complexity results and polynomial algorithms for $\Delta \leq 2$. Surprisingly, when k is unbounded and the network reduces to a single edge the problem remains *NP*-hard. Table 1 summarizes the results.

2 Definitions and Notation

For a graph G , we denote by V its node set, by E its edge set, and by Δ its maximum node degree. Unless otherwise stated, the graph $G = (V, E)$ representing the network is assumed to be undirected, connected, and without multiple edges and loops. A global assignment of the interfaces to the nodes in V is given in terms of an appropriate interface assignment function W , as follows.

Definition 1. A function $W: V \rightarrow 2^{\{1,2,\dots,k\}}$ is said to cover graph G if for each $\{u, v\} \in E$ we have $W(u) \cap W(v) \neq \emptyset$.

The cost of activating an interface i is given by the cost function $c: \{1, 2, \dots, k\} \rightarrow \mathbb{Z}_0^+$ and it is denoted as $c(i)$. The bandwidth allowed by a given interface i is defined by the bandwidth function $b: \{1, 2, \dots, k\} \rightarrow \mathbb{Z}_0^+$ and it is denoted as $b(i)$. It follows that each node holding an interface i pays the same cost $c(i)$ and provides the same bandwidth $b(i)$ by activating i . The considered *BCMI* optimization problem is formulated as follows.

BCMI: Bandwidth Constraints in Multi-Interface Networks

Input: A graph $G = (V, E)$, a source node $s \in V$, a target node $t \in V$, a set of interfaces $I = \{1, 2, \dots, k\}$, an allocation of available interfaces $W: V \rightarrow 2^I$ covering graph G , an interface cost function $c: I \rightarrow \mathbb{Z}_0^+$, an interface bandwidth function $b: I \rightarrow \mathbb{Z}_0^+$ and a bound $B \in \mathbb{Z}_0^+$.

Solution: An allocation of active interfaces $W_A: V \rightarrow 2^I$, $W_A(v) \subseteq W(v)$ for all $v \in V$ and a flow function $f: V \times V \times I \rightarrow \mathbb{Z}_0^+$ such that:

- $f(u, v, i) = 0$ if $\{u, v\} \notin E$ or $W_A(u) \cap W_A(v) = \emptyset$ for all $u, v \in V$ and $i \in I$
- $\sum_{v \in V: f(u, v, i) > 0} f(u, v, i) \leq b(i)$ for all $u \in V$ and $i \in I$
- $f(u, v, i) = -f(v, u, i)$ for all $u, v \in V$ and $i \in I$
- $\sum_{v \in V, i \in I} f(u, v, i) = 0$ for all $u \in V \setminus \{s, t\}$
- $\sum_{v \in V, i \in I} f(s, v, i) = \sum_{v \in V, i \in I} f(v, t, i) \geq B$

Goal: Minimize the total cost of the active interfaces, $c(W_A) = \sum_{v \in V} \sum_{i \in W_A(v)} c(i)$.

Note that we can consider two variants of the above problem: the parameter k can be considered as part of the input (this is called the *unbounded case*), or k may be a fixed constant (the *bounded case*). In both cases we assume $k \geq 2$, since the case $k = 1$ admits an obvious unique solution given by the shortest path connecting s to t of maximum bandwidth $b(1)$. The case where the cost function is constant for each interface is called the *unit cost* case.

3 Hardness and approximation

In this section we first prove that *BCMI* is *NP*-hard even in the restricted case of unit cost, fixed $k \geq 2$, and fixed $\Delta \geq 3$. We then prove that, unless $P = NP$, the problem is inapproximable within a factor of $\eta \log B$, for a certain constant η , or within a factor of $\Omega(\log \log |V|)$. Finally, we provide a polynomial time b_{\max} -approximation algorithm, where $b_{\max} = \max_{i \in I} b(i)$.

Theorem 1. *BCMI is NP-hard even when restricted to the unit cost interface case for any fixed $\Delta \geq 3$ and $k \geq 2$.*

Proof. We prove that the underlying decisional problem, denoted by *BCMI_D*, is in general *NP*-complete. We need to add one further bound $B' \in \mathbb{Z}_0^+$ such that

the problem will be to ask whether there exists an activation function which induces a total cost of the active interfaces of at most B' .

Given an allocation function of active interfaces for an instance of $BCMI_D$, to check whether the induced subgraph allows a flow bandwidth greater than or equal to B of total cost smaller than or equal to B' is linear in the number of edges of the input graph G . The proof then proceeds by a polynomial reduction from the well-known *Exact Cover by 3-Sets* problem. The problem is known to be *NP*-complete [9] and it can be stated as follows:

$X3C$: Exact Cover by 3-Sets

Input: Set X with $|X| = 3q$ and a collection C of 3-element subsets of X .
Question: Is there an exact set cover for X , i.e. a subset $C' \subseteq C$ such that $|C'| = q$ and every element of X belongs to exactly one member of C' ?

Given an instance of $X3C$, we construct an instance of $BCMI_D$ where the graph G consists of copies of subgraphs $N(\ell)$ and $T(\ell)$, $\ell \geq 1$ (see Fig. 1). Subgraph $N(\ell)$ consists of 3ℓ nodes $\{x_1, x_2, \dots, x_\ell\} \cup \{y_1, y_2, \dots, y_\ell\} \cup \{w_1, w_2, \dots, w_\ell\}$ and edges $\{x_i, x_{i+1}\}$, $\{w_i, w_{i+1}\}$, for $i = 1, 2, \dots, \ell - 1$ and $\{x_i, y_i\}$, $\{y_i, w_i\}$, for $i = 1, 2, \dots, \ell$. Subgraph $T(\ell)$ is a binary tree consisting of a complete binary tree BT with $2^{\lceil \log_2 \ell \rceil} - 1$ nodes, and ℓ nodes adjacent to the leaves of BT . These nodes are the only leaves of $T(\ell)$, i.e. every leaf of BT is connected to at least one leaf of $T(\ell)$. We call r the root of $T(\ell)$. Note that, each path from r to a leaf of $T(\ell)$ is constituted of $\lceil \log_2 \ell \rceil + 1$ nodes. Moreover, when $\ell = 1$, BT is empty and $T(\ell)$ consists of a single node.

We now define the graph G , see Fig. 1 right. Let s and t be two nodes of G . For each element C_i of C , $i = 1, 2, \dots, |C|$, G contains a node c^i , a copy of $N(3)$, denoted as $N^i(3)$ and a copy of $T(3)$, denoted as $T^i(3)$, with root r^i and leaves l_1^i, l_2^i, l_3^i . Vertices x_1^i and w_3^i of $N^i(3)$ are adjacent to c^i and r^i , respectively. All nodes c^i form a path P in G , that is $\{c^i, c^{i+1}\}$ is an edge of G , for $i = 1, 2, \dots, |C| - 1$. Node s of G is adjacent to c^1 , while node $c^{|C|}$ is adjacent to node x_1^0 belonging to a copy $N^0(1)$ of $N(1)$ with nodes x_1^0, y_1^0 and w_1^0 .

Let e_j , $j = 1, 2, \dots, 3q$, be the elements of X and let $\mu(e_j)$ be the number of sets $C_i \in C$ containing e_j , for each j . Let $\mu = \max_j \{\mu(e_j)\}$. For each element e_j , G contains a copy of $T(\mu)$, called $T^j(\mu)$, with root r^j , and a copy $N^j(1)$ of $N(1)$, with nodes x_1^j, y_1^j and w_1^j . Root r^j is adjacent to $x_1^j \in N^j(1)$, for each $j = 1, 2, \dots, 3q$. If e_j is in C_i , for some i and j , then there is an edge from a leaf of $T^i(3)$ to a leaf of $T^j(\mu)$. These edges are pairwise disjoint. Note that, even if each leaf of $T^i(3)$, $i = 1, 2, \dots, |C|$ is adjacent to a leaf in $T^j(\mu)$, for some $j \in \{1, 2, \dots, 3q\}$, the contrary is not true: there could be a leaf of $T^j(\mu)$, for some j , not adjacent to any leaf of $T^i(3)$, $i = 1, 2, \dots, |C|$.

G also contains a copy of $T(3q + 1)$, having the root adjacent to node t , and leaves adjacent to nodes w_1^j , $j = 0, 1, \dots, 3q$. The set of interfaces I is $\{1, 2\}$, with $c(1) = c(2) = 1$ and $b(1) = 1$, $b(2) = 3q + 1$. All the nodes in G have interface 2 apart from nodes labeled y in the copies of $N(1)$ and $N(3)$. All the nodes in the copies of $N(1)$ and $N(3)$ have interface 1: no further node in G has interface 1.

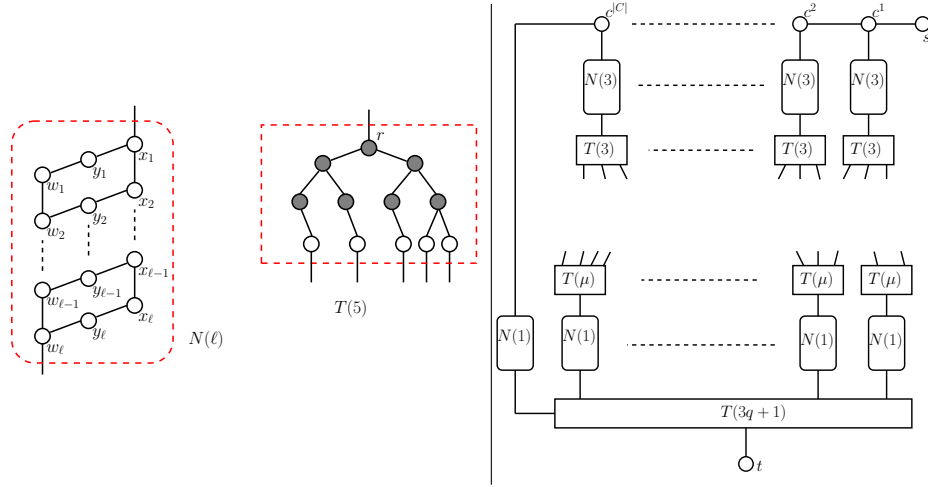


Fig. 1. Left: The subgraphs used in the proofs of Theorems 1 and 2. Right: The graph G in the transformation from $X3C$ to $BCMI_D$.

When all the interfaces of the nodes in copies of $N(\ell)$ ($T(\ell)$, resp.), for a certain $\ell \geq 0$, are active the total cost is 5ℓ ($2^{\lceil \log_2 \ell \rceil} - 1 + \ell$, resp.). In $T(\ell)$, when only the interfaces of the nodes in a single path from r to a leaf are active, the total cost is $\lceil \log_2 \ell \rceil + 1$. Let $B = 3q + 1$ and $B' = |C| + q(42 + 3\lceil \log_2 \mu \rceil) + 2^{\lceil \log_2(3q+1) \rceil} + 7$.

Assume that $X3C$ has a positive answer, i.e., there exists an exact set cover $C' = \{C_{i_1}, C_{i_2}, \dots, C_{i_q}\} \subseteq C$ for X . We show that also $BCMI_D$ has a positive answer, i.e., there exists an activation function W_A of the available interfaces such that the bandwidth allowed from s to t is bigger than or equal to B and the total cost is smaller than or equal to B' . Function W_A is defined as follows. Along with interfaces of nodes s, t , all the interfaces of nodes in $T(3q+1)$, $N^j(1)$, $j = 0, 1, \dots, 3q$, and c^i , $i = 1, 2, \dots, |C|$, are active. All the interfaces of nodes in $N^{i_j}(3)$ and $T^{i_j}(3)$, for each $C_{i_j} \in C'$, $j = 1, 2, \dots, q$, are active. Moreover, if $e_j \in X$ is covered by $C_i \in C'$, then all the interfaces of nodes in $T^j(\mu)$ belonging to the path from r^j to a leaf in $T^i(3)$ are active. No further interface is active. The flow function is defined as 1 in nodes y of active copies of $N(1)$ and $N(3)$ and in the remainder of G it is defined to satisfy the flow conservation constraints.

The total cost of active interfaces is given by 2, for nodes s and t ; $|C|$, for nodes $c^i \in P$, $i = 1, 2, \dots, |C|$; $15q + 6q$ for nodes in $N^{i_j}(3)$ and $T^{i_j}(3)$, $j = 1, 2, \dots, q$; $3q(\lceil \log_2 \mu \rceil + 1)$ for nodes in $T^j(\mu)$, $j = 1, 2, \dots, 3q$; $5(3q + 1)$ for nodes in $N^j(1)$, $j = 0, 1, \dots, 3q$; and $2^{\lceil \log_2(3q+1) \rceil} + 3q$ for nodes in $T(3q + 1)$. Summing up all the values we obtain a cost equal to B' .

Regarding the total bandwidth, note that a copy of $N(\ell)$ has a maximum bandwidth of ℓ . As $X3C$ has a positive answer, each element of X is covered, then the flow through each subgraph $N^j(1)$, $j = 1, 2, \dots, 3q$, is exactly 1. As all the interfaces in P are active, we also have another unit of flow from $N^0(1)$ that

reaches t through the $T(3q+1)$ subgraph, hence obtaining a total flow of $3q+1$, i.e., $BCMI_D$ has a positive answer.

Now, let us assume we have a positive answer to $BCMI_D$. As the total flow received by t is greater than or equal to $B = 3q+1$, there is a flow of value 1 in each subgraph $N^j(1)$, $j = 0, 1, \dots, 3q$, meaning that each element of X is covered. Let us suppose, by contradiction, that the flow reaching the $N^j(1)$, $j = 1, 2, \dots, 3q$ subgraphs, implies the activation of the interfaces in $q' > q$ subgraphs among the $N^i(3)$, $i = 1, 2, \dots, |C|$ copies of $N(3)$. In this case there will be q'_1 subgraphs having one unit of flow, q'_2 subgraphs having 2 units of flow, and q'_3 subgraphs having 3 units of flow such that $q'_1 + 2q'_2 + 3q'_3 = 3q$.

The total cost for the interfaces activation is: 2, for nodes s and t ; $|C|$, for nodes in P (all the interfaces in P are active as $N^0(1)$ receives one unit of flow); $7q'_1 + 11q'_2 + 15q'_3$ for nodes in $N^i(3)$; $6q$ for nodes in $T^i(3)$, $i = 1, 2, \dots, q$; $3q(\lceil \log_2 \mu \rceil + 1)$ for nodes in $T^j(\mu)$, $j = 1, 2, \dots, 3q$; $5(3q+1)$ for nodes in $N^j(1)$, $j = 0, 1, \dots, 3q$, and $2^{\lceil \log_2(3q+1) \rceil} + 3q$ for nodes in $T(3q+1)$.

Then the total cost is $|C| + q(27q + 3\lceil \log_2 \mu \rceil) + 2^{\lceil \log_2(3q+1) \rceil} + 7 + 7q'_1 + 11q'_2 + 15q'_3$. As $7q'_1 + 11q'_2 + 15q'_3 > 5(q'_1 + 2q'_2 + 3q'_3) = 15q$, the total cost is greater than B' , a contradiction. Hence there are exactly q subgraphs $N^{i_j}(3)$, $j = 1, 2, \dots, q$ with 3 units of flow each and the corresponding sets C_{i_j} , $j = 1, 2, \dots, q$, represent a solution for $X3C$. \square

Theorem 2. *BCMI cannot be approximated within a factor of $\eta \log B$, for a certain constant η , or within a factor of $\Omega(\log \log |V|)$, for any fixed $\Delta \geq 3$ and $k \geq 3$, unless $P = NP$.*

Theorem 2 also holds when the number of interfaces is unbounded. We now provide a b_{\max} -approximation algorithm for any instance of $BCMI$, where b_{\max} is the maximum bandwidth value among the interfaces in I . The algorithm consists in relaxing $BCMI$ to the well-known *Integral Minimum Cost Flow (IMCF)* problem [1]. In the proof of the next theorem, we transform an instance of $BCMI$ into an instance of $IMCF$, and we show that such a transformation guarantees an approximation factor of b_{\max} . Let \mathcal{A} be an algorithm which optimally solves $IMCF$ in a graph $H = (V', E')$ in polynomial time $P_{\mathcal{A}}(|V'| + |E'|)$.

Theorem 3. *There exists a polynomial time b_{\max} -approximation algorithm for $BCMI$ which requires $O(|V|k^2 + |E| + P_{\mathcal{A}}(|V|k^2 + |E|))$ time.*

Proof. First, we transform an instance I_1 on a graph $G = (V, E)$ of $BCMI$ in an instance of an equivalent problem defined on a directed graph $G' = (V', A)$ without using multiple interfaces but associating costs and bandwidth only to arcs in A . The particular instance I_2 of such problem is defined as follows. Informally, for each interface of each node, there is an arc which has the same cost and bandwidth of the considered interface. The head of each of such arcs is connected to the tail of another arc of the same kind if they share an interface or they represent different interfaces of the same node. Formally, there are two nodes in V' for each node in V and for each interface of each node:

$$V' = \{(\bar{v}, i), (\underline{v}, i) \mid v \in V, i \in W(v)\} \cup \{\tilde{s}, \tilde{t}\},$$

$A = \{((\bar{v}, i), (\underline{v}, i)) \mid v \in V, i \in W(v)\} \cup \{((\underline{v}, i), (\bar{v}, j)) \mid v \in V, i, j \in W(v) \text{ s.t. } i \neq j\} \cup \{((\underline{v}, i), (\bar{u}, i)) \mid \{u, v\} \in E, i \in W(v) \cap W(u)\} \cup \{(\tilde{s}, (\bar{s}, i)), ((\underline{t}, j), \tilde{t}) \mid i \in W(s), j \in W(t)\}$.

The capacity of each arc $a = ((\bar{v}, i), (\underline{v}, i))$ is set to $b'(a) = b(i)$ whereas the capacity of each other arc is unlimited. The cost $c'(a)$ of each arc $((\bar{v}, i), (\underline{v}, i))$ is set to $c(i)$ and it is 0 for the remaining arcs. The objective is to find a flow function which minimizes the overall cost of arcs with positive flow and guarantees a flow of B between \tilde{s} and \tilde{t} .

Given a solution for I_2 , which defines a flow function f_2 , we can define a solution for I_1 by assigning a flow function $f_1(v, u, i) = f_2((\underline{v}, i), (\bar{u}, i)) - f_2((\underline{u}, i), (\bar{v}, i))$, for each $v, u \in V$ and $i \in W(v) \cap W(u)$. Vice versa, given a solution for I_1 , which defines a flow function f'_1 , we can define a solution for I_2 by assigning a flow function f'_2 such that $f'_2((\underline{v}, i), (\bar{u}, i)) = f'_1(v, u, i)$, if $f'_1(v, u, i) > 0$ and $f'_2((\underline{v}, i), (\bar{u}, i)) = 0$ otherwise, for each $v, u \in V$ and $i \in W(v) \cap W(u)$. The flows in the remainder of A are set in order to satisfy flow conservation constraints. It is not difficult to note that the feasibility of f_2 (f'_1 , resp.) implies the feasibility of f_1 (f'_2). Moreover, the cost of f_2 (f'_1 , resp.) is equal to the cost of f_1 (f'_2) as the cost of arcs $((\bar{v}, i), (\underline{v}, i))$ in A is $c(i)$ and it is 0 for any other arc. By the above discussion it follows that we can solve I_1 by solving I_2 .

We find an approximate solution for I_2 by using an *IMCF* instance. The *IMCF* problem consists of finding an integral flow greater than or equal to a given quantity between two nodes in a directed graph H where each arc a has a capacity $\beta(a)$ and cost $\chi(a)$. The objective is to minimize the function $\sum_{a \in A^+} \chi(a) \cdot f(a)$, where $f(a)$ is the flow on arc a and A^+ is the set of arcs with positive flow. This problem admits a polynomial time algorithm (see, e.g., [15]).

We obtain an *IMCF* instance I_3 from I_2 by setting $H = G'$, $\beta(a) = b'(a)$, and $\chi(a) = c'(a)/b'(a)$, for each $a \in A$.

Let us denote as f^* and f^{IMCF} two optimal flow functions for I_2 and I_3 , respectively and as A^* and A^{IMCF} the corresponding sets of arcs with positive flow. By definition, $\text{OPT} = \sum_{a \in A^*} c'(a)$. As $f^*(a) \leq b'(a)$, it follows that

$$\sum_{a \in A^*} c'(a) \geq \sum_{a \in A^*} c'(a) \cdot \frac{f^*(a)}{b'(a)} = \sum_{a \in A^*} \chi(a) \cdot f^*(a).$$

By the optimality of A^{IMCF} it follows that

$$\sum_{a \in A^*} \chi(a) \cdot f^*(a) \geq \sum_{a \in A^{IMCF}} \chi(a) \cdot f^{IMCF}(a) = \sum_{a \in A^{IMCF}} \frac{c'(a)}{b'(a)} \cdot f^{IMCF}(a).$$

As $f^{IMCF}(a) \in \mathbb{Z}_0^+$, for each $a \in A$, then $f^{IMCF}(a) \geq 1$, for each $a \in A^{IMCF}$. Moreover, $b_{\max} \geq b'(a)$, for each $a \in A^{IMCF}$.

Therefore, $\sum_{a \in A^{IMCF}} \frac{c'(a)}{b'(a)} \cdot f^{IMCF}(a) \geq \frac{1}{b_{\max}} \sum_{a \in A^{IMCF}} c'(a)$. \square

Corollary 1. *Let $b \in \mathbb{Z}_0^+$. If $b(i) = b$ for each $i \in I$, BCMI is solvable within $O(|V|k^2 + |E| + P_{\mathcal{A}}(|V|k^2 + |E|))$.*

Proof. If $b = 1$, then the b_{\max} -approximation algorithm given in Theorem 3 optimally solves *BCMI*. Otherwise, it is enough to solve the problem with required bandwidth of $\bar{B} = \lceil \frac{B}{b} \rceil$ and bandwidth $\bar{b}(i) = 1$, for each interface i . \square

4 Particular cases, $\Delta \leq 2$

In this section, we consider graphs of bounded degree $\Delta \leq 2$. As announced in Table 1, we now prove that when the number of interfaces k is a given constant, the problem can be optimally solved in polynomial time. On the other hand, if k is unbounded, we show that the problem remains *NP*-hard.

For $\Delta \leq 1$, the input graph can be composed of either one single node or two nodes connected by one edge. In the first case, there are no interfaces to be activated, as the source and the destination coincide. In the second case, the problem already starts to be interesting.

Lemma 1. *BCMI is polynomially solvable within $O(1)$ time in the bounded case with $\Delta = 1$.*

Proof. *BCMI* can be solved by an exhaustive search among all the possible combinations of interfaces shared by s and t . The number of such combinations is $O(2^k)$. Among them, a resolution algorithm has to choose the cheapest one that guarantees at least B bandwidth. \square

For the unbounded case, i.e., when k is not a given constant, the same arguments of Lemma 1 do not apply to *BCMI* as the provided algorithm would show an exponential behavior. Surprisingly, in this setting the problem turns out to be already *NP*-hard by means of a simple polynomial transformation from the well known Knapsack problem. Indeed, we need to consider the so called Minimization Knapsack problem [11, 12].

MinKP: Minimization Knapsack

Input: An integer $d \in \mathbb{Z}_0^+$ and a set of n items, each one having weight $w_i \in \mathbb{Z}_0^+$ and profit $p_i \in \mathbb{Z}_0^+$, $i = 1, 2, \dots, n$.

Solution: An allocation of variables $y_i \in \{0, 1\}$, for $i = 1, 2, \dots, n$, such that $\sum_{i=1}^n w_i y_i \geq d$

Goal: Minimize $\sum_{i=1}^n p_i y_i$.

MinKP problem is the corresponding minimization version of the Knapsack problem. In other words, the goal is to minimize the profits of the items that remain out of the knapsack. If x_i , $i = 1, 2, \dots, n$, are the variables selecting the items for the classical knapsack problem and $c \in \mathbb{Z}_0^+$ its capacity, then the problem can be solved by means of *MinKP*, by setting $d = \sum_{i=1}^n w_i - c$ and $y_i = 1 - x_i$, $i = 1, 2, \dots, n$.

When $\Delta = 1$, that is when the input graph G consists of a single edge from s to t , the required solution must select a subset of interfaces among the ones shared by s and t in such a way that a bandwidth of B is guaranteed, and the cost for activating such interfaces is minimized. Intuitively, this particular case of *BCMI* is equivalent to the *MinKP* problem.

Theorem 4. *BCMI is polynomially equivalent to MinKP in the unbounded case with $\Delta = 1$.*

Proof. We have to show that there exist two polynomial time algorithms \mathcal{A} and \mathcal{B} such that, for each instance I_1 of *MinKP*, $\mathcal{A}(I_1)$ returns an instance I_2 of *BCMI*, for any solution σ' of I_2 , $\mathcal{B}(\sigma') = \sigma$ is a solution for I_1 , and the values of solutions σ and σ' are equal. Moreover, we have to show that there exist two polynomial time algorithms \mathcal{A}^{-1} and \mathcal{B}^{-1} such that, for each instance I_2 of *BCMI*, $\mathcal{A}^{-1}(I_2)$ returns an instance I_1 of *MinKP*, for any solution σ of I_1 , $\mathcal{B}^{-1}(\sigma) = \sigma'$ is a solution for I_2 , and the values of solutions σ and σ' are equal.

We now show the first part of the above statement by defining the polynomial algorithms \mathcal{A} and \mathcal{B} . Given an instance I_1 of *MinKP*, we consider an instance I_2 of *BCMI* made of nodes s and t , edge $\{s, t\}$ and, for each item i of I_1 , an interface shared between s and t with cost $c(i) = \frac{1}{2}p_i$ and bandwidth $b(i) = w_i$. Moreover, let $k = n$ and $B = d$. Note that, if, for some i , p_i is an odd number, we can scale all the profits p_i of a factor 2 in order to have $c(i) \in \mathbb{Z}_0^+$ for each $i = 1, 2, \dots, n$. This does not affect the generality of the proof as it is enough to divide by 2 the objective function value of the solution for I_1 which will be defined in the following. A feasible solution for I_2 selects a set of interfaces W , by means of an activation function, in such a way that $B \leq \sum_{i \in W} b(i)$. As $d = B \leq \sum_{i \in W} b(i) = \sum_{i \in W} w_i$ and the cost of activating interfaces W in both s and t is $2 \sum_{i \in W} c(i) = \sum_{i \in W} p_i$ we can define algorithm \mathcal{B} as the algorithm which selects items W in order to output a solution for I_1 . Finally, both \mathcal{A} and \mathcal{B} are polynomial time algorithms. This proves the first part of the theorem. For the second part of the theorem, it is enough to note that algorithms \mathcal{A} and \mathcal{B} can be naturally inverted. \square

Corollary 2. *BCMI is NP-hard in the unbounded case with $\Delta = 1$.*

Corollary 3. *In the unbounded case with $\Delta = 1$, BCMI admits a $(1 + \epsilon)$ -approximation algorithm which requires $O(\frac{k^2}{\epsilon})$ time, for any $\epsilon > 0$.*

Proof. It follows by applying the linear time algorithm \mathcal{A} of Theorem 4 which requires $O(k)$ time, and the algorithm from [10] which provides a $(1 + \epsilon)$ -approximation for *MinKP* in $O(\frac{k^2}{\epsilon})$ time. \square

For $\Delta = 2$, the input graph of *BCMI* is either a path or a cycle. Clearly, from Corollary 2, *BCMI* remains NP-hard in the unbounded case. The following theorems give polynomial time algorithms for the bounded case, and a refined approximation algorithm for paths in the unbounded case.

In the remainder, for a set of interfaces W , we denote as $c(W)$ the cost of activating the interfaces in W , formally: $c(W) = \sum_{i \in W} c(i)$.

Theorem 5. *BCMI is solvable within $O(|V|)$ time in the bounded case when the input graph is a path.*

Theorem 6. *In the unbounded case, if the input graph is a path, BCMI admits a $(2 + \epsilon)$ -approximation algorithm which requires $O(|V|\frac{k^2}{\epsilon})$ time, for any $\epsilon > 0$.*

Proof. Let us denote the input path as a sequence of n nodes: $s \equiv x_0, x_1, \dots, x_{n-1} \equiv t$. We define an algorithm \mathcal{C} as follows. It defines $n-1$ *MinKP* problems, each one arising from one different edge $e_i = \{x_{i-1}, x_i\}$ of the path, $1 \leq i \leq n-1$, by using the linear time algorithm \mathcal{A} of Theorem 4. From Corollary 3, this implies that for each e_i and for any $\epsilon > 0$, a $(1 + \epsilon)$ -approximation for *MinKP* can be guaranteed. Algorithm \mathcal{C} chooses, for each $1 \leq i \leq n-1$, interfaces W_i arising from the approximate solution of the related knapsack problem on edge e_i , that is interfaces W_i are activated on nodes x_{i-1} and x_i .

For each $1 \leq i \leq n-1$, let us denote as W_i^* , the sets of active interfaces in nodes x_{i-1} and x_i covering edge e_i for an optimal solution of *BCMI* for the input path; and let W_i^{MK} the sets of active interfaces in nodes x_{i-1} and x_i covering edge e_i for an optimal solution of the *MinKP* problem obtained by \mathcal{C} for the input path.

Note that, for some i , the set $W_i \cap W_{i+1}$ is not necessarily empty, which means that node x_i uses a set of interfaces for communicating both with x_{i-1} and x_{i+1} . Thus, in this case, the cost paid for activating the interfaces used by x_i is less than $c(W_i) + c(W_{i+1})$ and the same holds for solutions W_i^* and W_i^{MK} . It follows that, for each $1 \leq i \leq n-1$ the cost paid for activating interfaces in W_i in nodes x_i and x_{i-1} is at most $2c(W_i)$ and the overall cost of the solution provided by \mathcal{C} is less than or equal to $2 \sum_{i=1}^{n-1} c(W_i)$. As from Corollary 3 we are using in each edge a $(1 + \epsilon)$ -approximation algorithm for the knapsack problem, it follows that: $2 \sum_{i=1}^{n-1} c(W_i) \leq 2 \sum_{i=1}^{n-1} (1 + \epsilon) c(W_i^{MK})$. As W_i^{MK} is an optimal solution for *MinKP* on edge e_i which guarantees a bandwidth of B , $c(W_i^{MK}) \leq c(W_i^*)$, for each $1 \leq i \leq n-1$, and hence: $2(1 + \epsilon) \sum_{i=1}^{n-1} c(W_i^{MK}) \leq 2(1 + \epsilon) \sum_{i=1}^{n-1} c(W_i^*) \leq 2(1 + \epsilon) \left(\sum_{i=1}^{n-2} c(W_i^* \cup W_{i+1}^*) + c(W_{n-1}^*) \right) \leq 2(1 + \epsilon) \text{OPT}$, where the two last inequalities follow from the fact that in an optimal solution the cost of activating interfaces for each node x_i is $c(W_i^* \cup W_{i+1}^*) \geq c(W_i^*)$ and the overall cost is $\text{OPT} = c(W_1^*) + \sum_{i=1}^{n-2} c(W_i^* \cup W_{i+1}^*) + c(W_{n-1}^*)$.

The complexity of \mathcal{C} is $O(n \frac{k^2}{\epsilon})$ as it is composed of $n-1$ executions of algorithm \mathcal{A} of Theorem 4 which requires $O(k)$ time, and $n-1$ executions of algorithm from [10] which requires $O(\frac{k^2}{\epsilon})$ time. By defining $\epsilon' = 2\epsilon$, Algorithm \mathcal{C} provides a $(2 + \epsilon')$ -approximated solution and requires $O(|V| \frac{k^2}{\epsilon'})$ time. \square

When the input graph is a cycle, since there are two paths from s to t , it is not always clear how the bandwidth B must be split among the two possible ways. However, the following theorem can be stated for the bounded case.

Theorem 7. *BCMI is solvable within $O(|V|)$ time in the bounded case when the input graph is a cycle.*

5 Conclusion

We have considered the Bandwidth Constraints in Multi-Interface Networks problem. We focused on problem hardness and approximation factors in general

and more specific settings. The obtained results have shown that the problem is *NP*-hard to be optimally or approximately solved. Polynomial algorithms for special cases have been provided. Further investigation for better performing approximation algorithms or heuristics remain challenging problems. Another interesting issue is to study the problem from a distributed point of view.

References

1. R. Ahuja, T. Magnanti, and J. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, 1993.
2. S. Athanassopoulos, I. Caragiannis, C. Kaklamanis, and E. Papaioannou. Energy-efficient communication in multi-interface wireless networks. In *34th Int. Symp. on Mathematical Foundations of Computer Science (MFCS)*, volume 5743 of *LNCS*, pages 102–111. Springer, 2009.
3. P. Bahl, A. Adya, J. Padhye, and A. Walman. Reconsidering wireless systems with multiple radios. *SIGCOMM Comput. Commun. Rev.*, 34(5):39–46, 2004.
4. M. Caporuscio, D. Charlet, V. Issarny, and A. Navarra. Energetic Performance of Service-oriented Multi-radio Networks: Issues and Perspectives. In *6th Int. Workshop on Software and Performance (WOSP)*, pages 42–45. ACM Press, 2007.
5. D. Cavalcanti, H. Gossain, and D. Agrawal. Connectivity in multi-radio, multi-channel heterogeneous ad hoc networks. In *IEEE 16th Int. Symp. on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pages 1322–1326. IEEE, 2005.
6. G. D’Angelo, G. Di Stefano, and A. Navarra. Minimizing the Maximum Duty for Connectivity in Multi-Interface Networks. In *Proceedings of the 4th Annual International Conference on Combinatorial Optimization and Applications (COCOA)*, volume 6509 Part II of *LNCS*, pages 254–267. Springer, 2010.
7. R. Draves, J. Padhye, and B. Zill. Routing in multi-radio, multi-hop wireless mesh networks. In *10th annual international conference on Mobile computing and networking (MobiCom)*, pages 114–128. ACM, 2004.
8. A. Faragó and S. Basagni. The effect of multi-radio nodes on network connectivity—a graph theoretic analysis. In *IEEE Int. Workshop on Wireless Distributed Networks (WDM)*. IEEE, 2008.
9. M. R. Garey and D. S. Johnson. *Computers and Intractability, A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, New York, 1979.
10. G. Gens and E. Levner. Computational complexity of approximation algorithms for combinatorial problems. In *8th Int. Symp. on Mathematical Foundations of Computer Science (MFCS)*, volume 74 of *LNCS*, pages 292–300. Springer, 1979.
11. S. Görtz and A. Klose. Analysis of some greedy algorithms for the single-sink fixed-charge transportation problem. *Journal of Heuristics*, 15(4):331–349, 2009.
12. M. Güntzer and D. Jungnickel. Approximate minimization algorithms for the 0/1 knapsack and subset-sum problem. *Operations Research Letters*, 26(2):55–66, 2000.
13. R. Klasing, A. Kosowski, and A. Navarra. Cost Minimization in Wireless Networks with a Bounded and Unbounded Number of Interfaces. *Networks*, 53(3):266–275, 2009.
14. A. Kosowski, A. Navarra, and M. Pinotti. Exploiting Multi-Interface Networks: Connectivity and Cheapest Paths. *Wireless Networks*, 16(4):1063–1073, 2010.
15. E. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart, and Winston, 1976.