Complexity and approximability issues of Shared Risk Resource Group

David Coudert, Pallab Datta, Stéphane Pérennes, Hervé Rivano, Marie-Emilie Voge

Complexity and approximability issues of Shared Risk Resource Group

D. Coudert — P. Datta — S. Perennes — H. Rivano — M.-E. Voge

N° 5859

Rapport de recherche

INRIA

SOPHIA ANTIPOLIS

Complexity and approximability issues of Shared Risk Resource Group

D. Coudert , P. Datta , S. Perennes , H. Rivano , M.-E. Voge

Thème COM — Systèmes communicants
Projet Mascotte, commun I3S-CNRS/INRIA/UNSA

Abstract: This article investigates the consequences of the Shared Risk Ressource Groups (SRRG) model on classical combinatorial concepts of network survivability. It focuses on complexity and approximability issues, and on the evolutions of the relationships among these questions.

We introduce a combinatorial model for SRRG based on edge-colored graphs. The notions of colored cut and colored spanning tree are introduced, the hardness and non-approximability of path, cut and spanning tree colored optimization problems are proved. We provide approximation algorithms, and investigate specific polynomial cases.

Deep differences between colored combinatorial questions and their counterparts in classical graph theory are shown. In particular, results concerning the relationships among colored problems are presented.

Key-words: Reliability, Shared Risk Resource Group, colored graphs, complexity, approximability.

Complexité et approximabilité
des groupes de ressources partageant un risque

Résumé : Dans ce rapport, nous étudions l'influence de la modélisation des groupes de ressources partageant un risque (SRRG) sur les concepts classiques d'optimisation combinatoire dans les réseaux. Nous concentrons nos travaux sur les questions de complexité et d'approximabilité, ainsi que sur l'évolution des relations entre ces problèmes.

Nous introduisons un nouveau modèle combinatoire pour les SRRG basé sur des graphes arêtes colorés. Nous introduisons les notions de coupe coloré et d'arbre couvrant coloré puis nous prouvons la complexité et l'inapproximabilité des problèmes de chemins, coupe et arbres couvrant colorés. Ensuite, nous proposons des algorithmes d'approximations et nous étudions des cas polynomiaux spécifiques.

Nous exhibons les différences fondamentales entre les problèmes combinatoires colorés et leur contrepartie en théorie des graphes classique. En particulier nous présentons les relations entre les différents problèmes colorés.

Mots-clés : Fiabilité, Groupes de ressources partageant un risque, graphes coloré, complexité, approximabilité

1   Introduction

In graph theory many problems consists in finding a minimum cost subset of edges satisfying some given property. The cost is often the size of the set, or a linear function such as the sum of the cost of the edges. Problems such as minimum cost spanning tree, minimum cut problems and shortest paths belong to this familly. These problems model applications in which edges are independent.

Recently the need to take into account correlations between edges has appeared, motivated by the network survivability concept of *Shared Risk Link Group*, later generalized to *Shared Risk Resource Group* (see section 2). A SRRG is a set of resources that will break down simultaneously if a given failure occurs. One can then define the "*safest*" path as the one using the least number of groups, and two "*disjoints*" paths as a pair of paths using resources belonging to different groups.

Shared Risk Resources Groups are naturally modeled by associating to each risk a *color*, and to each resource an edge colored by each of the colors representing the risks affecting the resource. Henceforth, one can restate classical graph problems as follows. Given a graph and a set of colors on the edges, find a minimum cost subset of the colors having some property. As example, one may wish to find the path connecting two vertices that uses the minimum number of colors, such a path is indeed the "safest" one, as previously defined.

This work is motivated by the consequences of the concept of SRRG, and its colored graph model, on combinatorial problems yielded by classical network survivability issues.. Therefore, we investigate the complexity and approximability of subgraph problems that are polynomial in the usual settings, such as paths, cuts and spanning tree problems. We show that these problems become NP-Hard and hard to approximate, unless strong restrictions are assumed. We also show that the relationships between colored graphs problems are deeply different from those between their counterparts in classical graph theory.

2   Shared Risk Resource Groups

Shared Risk Resource Groups are relevant in numerous practical settings. Indeed, it can model correlated traffic jams of road networks, as well as cascading failure scenario in electrical networks. In particular, SRRG have been introduced for modeling group of resources in a telecommunication network which may fail simultaneously [11, 13, 14].

Multiple failures scenario has been neglected for a long time by survivable network design studies. Recently, this kind of situation has revealed unavoidable with the spread of modern day multilayer networks such as IP/WDM, MPLS networks, P2P or GRID computing overlay structures. As example, in such a multilayer network, multiple and apparently independent virtual links may be paths sharing a link on the underlying network, as illustrated in Fig. 1. Consequently, a single failure on the link (FG) would induce multiple failures on the set of links of the virtual topology.

In general cases, a Shared Risk Resource Group of a network may include links and vertices. Using the usual graph representation of the network, a SRRG would thus be a
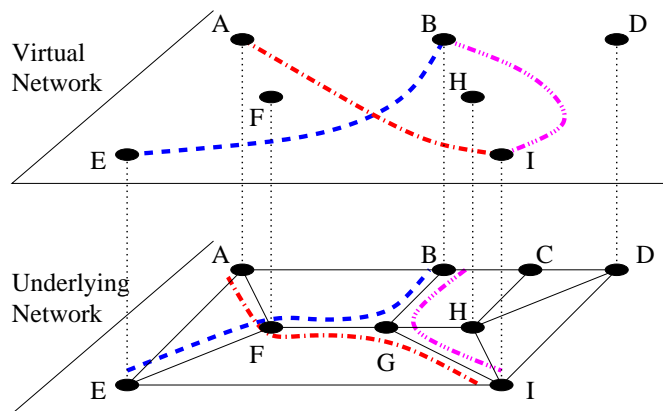
Figure 1: A Shared Risk Link Group : (AI) and (BE) links share the same risk (FG).

set of vertices and edges. However, using straightforward graph transformations, one can consider only sets of edges without loss of generality.

Therefore, SRRG problems have mainly been studied in the context of network survivability. The *diverse routing problem* has been investigated from several viewpoints. It consists in finding two paths between a pair of vertices in a graph, such that no SRRG failure cause both paths to fail. This problem is much more difficult than the traditional edge or vertex disjoint path problem in graph theory [1, 7] because two links may belong to the same SRRG, independently of the topology of the graph. Furthermore, the generalized SRRG diverse routing problem is NP-complete [7].

A relaxation of this problem, the *minimum overlapping color disjoint paths problem*, as well as the *minimum cost SRRG diverse routing problem* and the *routing problem under both link capacity and path length constraints* are also NP-complete [7]. Consequently, several heuristics approaches have been studied [1, 9]. These studies have identified the existence of *trap topologies* in on-line algorithms [9, 16]: if a path crosses such a trap topology, there may not exist a diverse path, even though two diverse paths exist in the network.

A weighted version of these problems has also been investigated. If each SRRG has a probability of failure, the objective is to find one or more paths for each connection, such that the reliability for each connection is maximized [17].

3   Colored Graphs

In this section we define formaly *colored graphs*. However, we prove that it is enough to consider that edges are monochromatic.

According to the general case of Shared Risk Ressource Groups, an edge may belong to several colors, modeling the fact that a resource may face different and independent risks.

(a) A colored graph   (b) color of span 2   (c) color of span 1

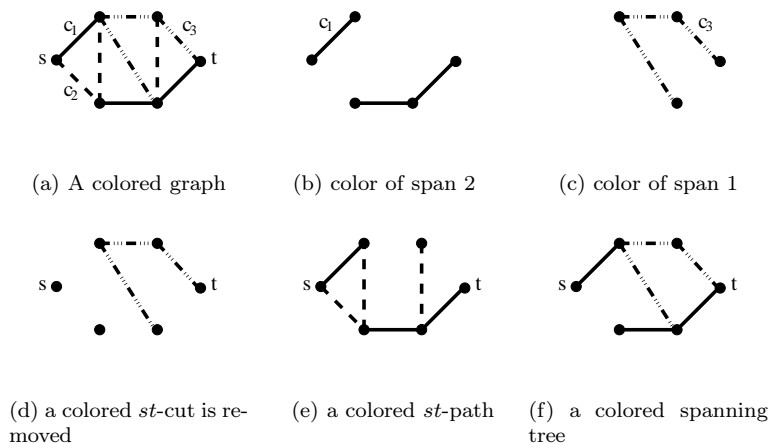(d) a colored st-cut is removed   (e) a colored st-path   (f) a colored spanning tree

Figure 2: Exemples

Depending on the context, this property may be interpreted in several ways whether *we have to choose one of the colors to cross the edge* or *we need all the colors to cross the edge*. However, in case we need to choose one of the colors, we can easily replace an edge belonging to X colors by X monochromatic parallel edges, and in case we need all the colors, we can replace the edge by a chain of X monochromatic edges.

Therefore, in the following we will assume that monochromatic edges, which means edge disjoint colors, Hence the following definitions.

Definition 1 (Colored graph) *A colored graph is a triple $G = (V, E, \mathcal{C})$ where (V, E) is an undirected graph and $\mathcal{C}$ is a partition of E. A weighted colored graph is a colored graph such that each color $c \in \mathcal{C}$ has a weight $w_c \geq 0$.*

We will also say that a vertex is adjacent to a color whenever an edge of that color is adjacent to that vertex. The colored degree of a vertex is then its number of adjacent colors.

Definition 2 (span of a color) *The span of a color is the number of connected components of the subgraph induced by the edges of this color (Fig. 2(b), 2(c)).*

The counterparts of classical problems consists in finding a minimum cost set of colors satisfying a given property. As exemple a **colored path** is defined as a subset of colors whose corresponding edge set contains an usual path (Fig. 2). Similarly, we define **colored st-cut**, **colored cut**, and **colored spanning tree**, and the following problems, motivated by network survivability issues.

The **Minimum Color Path problem** (MC-Path) consits in connecting two vertices using as few colors as possible. The weighted case is related to finding a path of minimum failure probability. The **Color Disjoint Paths problem** arises naturally. When no such set exists, one can look for **Minimum Overlapping Paths**, that is a set of paths sharing a minimum number of colors. Finding pairs of such paths is of specific interest. The corresponding problems are denoted 2-Color Disjoint Paths and 2-Min. Overlapping Paths. **MC-st-Cut problem** consists in finding a minimum size set of colors disconnecting two vertices s and t. Similarly, we can define the **MC-Multi-Cut** and **MC-Cut** problems. Note that the size of a MC-Cut is upper bounded by the minimum colored degree, as in classical graphs. The **MC-Spanning Tree problem** is to find a minimum size set of colors inducing a connected subgraph, as depicted in Fig. 2.

Table 1 sums up the results obtained for colored problems. The weak relations relations between Color Disjoint Paths and MC-st-Cut (analogous to min-cut equal max number of edge disjoint paths) are investigated in section 7.

4   MC-Path & MC-st-Cut

The complexity of the MC-st-Cut problem is similar to the one of the MC-Path problem [18], and is proven by a reduction to the Minimum Set Cover problem. In the remaining of the article, a **Minimum Set Cover instance** (Min. Set Cover) is defined by a set $U = \{u_1, \ldots, u_n\}$ and a collection $S = \{s_1, \ldots s_m\}$ of subsets of U. The problem is then to find a minimum size subcollection $S' \subseteq S$ such that each element of U is contained in at least one of its sets [4].

In the reduction, to each subset of Min. Set Cover corresponds a color of MC-st-Cut and to each element u_i of Min. Set Cover corresponds an st-path of MC-st-Cut which is colored according to the subsets where u_i belongs (Fig. 3).

Theorem 3 *The MC-st-Cut problem is NP-Hard.*

Actually, the aforesaid reductions to Min. Set Cover of MC-Path and MC-st-Cut give more than their NP-Hardness.

Theorem 4 *The MC-Path and MC-st-Cut problems are not approximable within a factor $o(\log |V|)$ unless $NP \subseteq TIME(n^{O(\log \log n)})$.*

A stronger innaproximability result can be derived, we provide the proof for MC-Path but the idea extends with syntaxical modification to the case of the MC-st-Cut.

Theorem 5 *For any constant $p \in \mathbb{N}^+$ the MC-Path and MC-st-Cut problems are hard to approximate within a factor $O(\log^p |V|)$ unless $NP \subseteq TIME(n^{O(\log \log n)})$.*

Table 1: Complexity and approximability properties of colored problems.

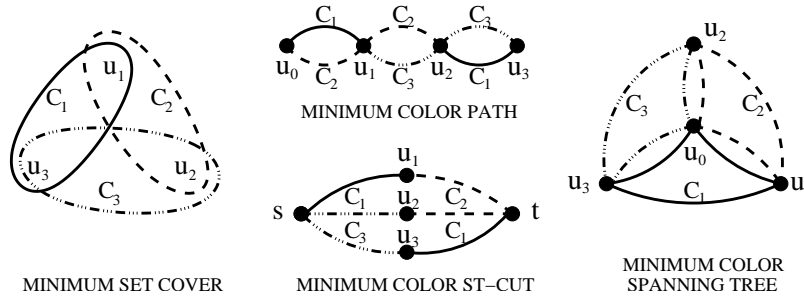|		MC-Path	2 Col. Disjoint Paths & 2 Min. Overlapping Paths	MC-st-Cut & MC-Multi-Cut	MC-Cut	MC-Spanning Tree
general	complexity	NP-Hard [18]	NP-Hard [18]	NP-Hard	?	NP-Hard
	non approx	$\forall p\ O(\log^p(n))$	?	$\forall p\ O(\log^p(n))$	?	$o(\log(n))$
	approx	?	?	?	?	$O(\log(n))$
size k	complexity	NP-Hard	?	NP-Hard	?	NP-Hard
	non approx	$\exists \varepsilon > 0\ k^\varepsilon$	?	$\exists \varepsilon > 0\ k^\varepsilon$	?	$o(\log(n))$
	approx	k	?	k	?	$O(\log(n))$
size 1	complexity non approx approx	P [2]	P	P	P	NP-Hard $o(\log(n))$ $O(\log(n))$
bounded degree		–	–	P	P	–

Figure 3: Some colored problems instances built from a Min. Set Cover instance.

The key of the proof of this theorem is the following self-improvement construction which is similar to the one used in an inapproximability proof for the Maximum Clique problem [6].

Given a colored graph $G = (V, E, \mathcal{C})$ and two vertices $a, b \in V$ we build a *square colored graph* $G^2_{a,b}$ (Fig. 4). First we replace each edge (u, v) of G by a copy G_{uv} of the graph G, the vertices u and v are identified with the vertices a and b of the copy. There are thus $|E|^2$ edges in $G^2_{a,b}$. Then for an edge (x, y) belonging to a copy G_{uv}, we define its color as (c_{uv}, c_{xy}), where c_{xy} is the color of edge (x, y) in G. Hence the color set of $G^2_{a,b}$ is $\mathcal{C} \times \mathcal{C}$.

Note that two edges belonging to distinct copies of G replacing edges of distinct colors of G are of distinct colors in $G^2_{a,b}$ too. In the same way the graph $G^p_{a,b}$ is built for any $p \in \mathbb{N}$ by replacing each edge of $G^{p-1}_{a,b}$ by a copy of G.

The proof of Theorem 5 derives easily from the two following lemmas.

Lemma 6 *Let G be a colored graph, a and b two vertices of G and $p \in \mathbb{N}$. Optimal solutions of MC-Path in G and in the graph G^p_{ab} of G satisfy the relation $|\mathrm{OPT}(\mathrm{G}^\mathrm{p}_\mathrm{ab})| = |\mathrm{OPT}(\mathrm{G})|^\mathrm{p}$.*
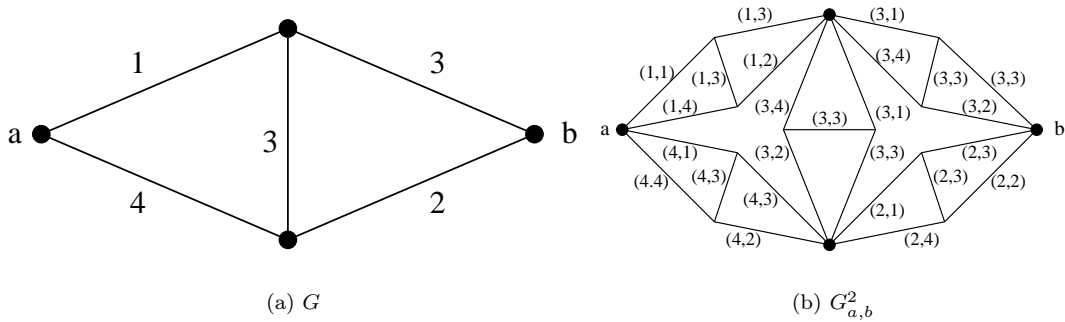

(a) G (b) $G^2_{a,b}$

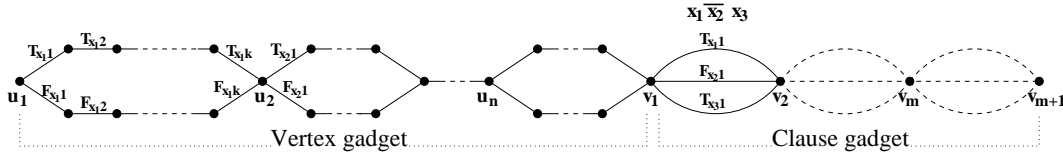Figure 4: A graph G and its square graph $G^2_{a,b}$.

Figure 5: Reduction of MC-Path with colors of span 2 to Maximum 3 Satisfiability

Lemma 7 *Let $\mathcal{A}$ be an algorithm for MC-Path, then $\forall p \in \mathbb{N}$ a colored path P_G of G of cost lower than $|\mathcal{A}(G^p_{ab})|^{\frac{1}{p}}$ can be deduced from $\mathcal{A}(G^p_{ab})$.*

Lemma 6 and 7 are based on the decomposition of a path of G^p_{ab} into paths of G. To prove Theorem 5 assume that $\mathcal{A}$ is a $O(\log^p |V|)$-approximation algorithm and run it on G^{p+1}_{ab}. Then Lemma 6 and 7 lead to a contradiction with Theorem 12.

4.1 Bounded color span case

In this section, we assume that the span of each color is bounded by a constant k. When $k = 2$, the reductions to Min. Set Cover of [18] and Theorem 3 are inadequate to show the NP-hardness of MC-Path and MC-*st*-Cut since they require that the maximum size of the subsets available to cover the elements of the Min. Set Cover instance is bounded by $k = 2$ too. In this case Min. Set Cover can be polynomialy solved by matching techniques. We give a reduction to Maximum 3 Satisfiability (M3S) which proves not only the NP-hardness of MC-Path and MC-*st*-Cut with colors of span at most two, but also an inapproximability property.

Again, the proofs are identical for MC-*st*-Cut and MC-Path thus we focus on the path problem. The idea of the reduction is to assign a set of colors to each literal of a M3S instance and construct a graph composed of two parts (Fig. 5). To cross the first part several paths are available, each one corresponds to an assignement (true/false) of the variables of M3S according to the colors it uses. The second part represents the clauses of M3S, to cross this part an additional color is needed for each unsatisfied clause of M3S. Of course the colors are assigned such that no color has span greater than two.

Proposition 8 *The MC-Path and MC-st-Cut problems are NP-Hard when the span of a color is bounded by two and are hard to approximate within a factor $1 + \varepsilon_1$ for some $\varepsilon_1 > 0$.*

Proposition 8 is the building bloc to prove Proposition 9. In the following, just replace *path* by *st-cut* to obtain the MC-*st*-Cut case.

Proposition 9 *When color have span at most k, MC-Path and MC-st-Cut are hard to approximate within a factor k^{ε_2}, for some $\varepsilon_2 > 0$.*

Proof: We construct from an instance of MC-Path on a graph G such that each color has span at most two, the instance G^p_{ab} for some $p \in \mathbb{N}$. Note that in G^p_{ab} each color has span at most 2^p. By lemma 6 and 7 no algorithm can approximate MC-Path within a factor

$(1 + \varepsilon_1)^p$ otherwise contradicting Proposition 8. Assume now that the span of a color is bounded by $k \in \mathbb{N}$. We can write $k = 2^{\log_2 k}$. Hence in the case $p = \lceil \log_2 k \rceil$, it is hard to approximate MC-Path within a factor $(1 + \varepsilon_1)^{\log_2 k} \leq (1 + \varepsilon_1)^{\lceil \log_2 k \rceil}$. This expression can be written $k^{\varepsilon_2} = (1 + \varepsilon_1)^{\log_2 k}$ for $\varepsilon_2 = (\log_2 k)(\ln(1 + \varepsilon_1))/\ln k$, that is $\varepsilon_2 = \log_2(1 + \varepsilon_1)$ is independent of k. □

5   Minimum Color Spanning Tree

Unlike other colored problems, when each color has span one the MC-Spanning Tree problem remains NP-Hard. This problem is very similar to Min. Set Cover. Indeed, we show that MC-Spanning Tree is as complex as set cover, and that a greedy algorithm achieves a logarithmic factor approximation.

Theorem 10 *Set cover is equivalent to a particular case of MC-Spanning Tree in which each color has span one.*

Sketch of proof: The reduction is based on associating a color of MC-Spanning Tree to each subset of Min. Set Cover. By forcing the existence of an element contained in every subset of the Min. Set Cover instance, on can ensure the equivalence between a cover of Min. Set Cover and a colored spanning tree of MC-Spanning Tree. □

The similarity between Min. Set Cover and MC-Spanning Tree is even deeper: an approximation algorithm [15] for Min. Set Cover can easily be adapted to MC-Spanning Tree.

Theorem 11 *A greedy $\mathcal{H}_{|V|-1}$-approximation algorithm exists for MC-Spanning Tree, where $\mathcal{H}_{|V|-1} = 1 + \frac{1}{2} + \cdots + \frac{1}{|V|-1} = O(\log |V|)$.*

Sketch of proof: The proof is based on the following greedy algorithm.

Each color c as a weight w_c. Start with the colored graph deprived of its edges. It contains thus $|V|$ connected components, each reduced to a single vertex. At iteration i, let n_c^i be the decrease of the number of connected components that would occur during this iteration if the edges of color c are added to the graph. The *distributed cost* of color c is then its weight distributed among the n_c^i removals, that is w_c/n_c^i. The edges of the color of lowest distributed cost are then added to the graph. The algorithm stops when the graph is connected.

By construction, the color set chosen by this greedy algorithm is a colored spanning tree. Note that there are at most $\min\{|V|, |\mathcal{C}|\}$ iterations.

The principle of the proof of Theorem 11 is to bound the distributed cost of a chosen color, that is the cost of each removal. □
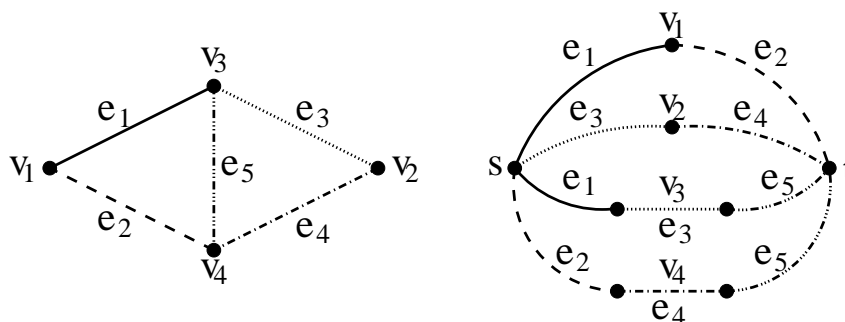
Figure 6: Example of edge-colored graph built from a Max. Independent Set instance.

6   Color Disjoint path

The counterpart of the k edge-disjoint paths problem consists in finding k color-disjoint paths. In the case of paths connecting two nodes (s, t-paths), the classical problem is easy to solve using single commodity flow techniques, or Menger augmentations. However, the colored version of the problem turns out to be much harder.

Theorem 12 *The Maximum Number of Color Disjoint Paths Problem is not approximable within a factor $o(n^{1-\varepsilon})$ unless $NP = P$.*

Proof: Our reduction uses Maximum Independent Set inapproximability. To a graph $G = (V, E)$, we associate the following color disjoint paths problem. We connect a source s to each vertex of G. Then each vertex v of G is connected to a sink t using a path whose length is the degree of v, as depicted in Fig. 6. The edges of the path are colored with the edges of G that are adjacent to v. One can easily see that two paths $u - t, v - t$ are color disjoint if u and v are not adjacent. The maximum number of color disjoint paths is henceforth the size of the maximum independent set of G.   □

7   Relations between problems

We will now show through examples that colored problems differ from classical graph theory, not only by their complexities, but also by their mutual relationships. It is in particular the case of the number of color disjoint st-paths and the MC-st-Cut, which are no longer tightly linked. The max flow-min cut relation being essential in the study of network reliability, this lack of link makes the Color Disjoint Paths problem even more thorny and stresses the interest of the Min. Overlapping Paths problem [18].

The sizes of a MC-Spanning Tree and a MC-Cut do not verify the same inequalities as their classical counterpart [5]. This is another example of these differences.

Theorem 13 *For each $k \in \mathbb{N}$, there is a colored graph and two vertices s and t of it such that the MC-st-Cut has value k while there cannot be found two color disjoint st-paths.*

Furthermore, this colored graph is also such that the MC-Cut has value k while there cannot be found two color disjoint spanning trees.

Proof: Let $k \in \mathbb{N}$ be a constant. We construct a colored graph G containing $\binom{2k}{k} + 1$ vertices and $2k$ colors. G is a path with multiple edges such that each subset of k colors is represented on exactly one of the multiple edges. Let s and t be the extremities of the path. A colored spanning tree in G is merely a colored st-path and a MC-Cut is actually a MC-st-Cut.

Suppose we have already chosen a subset C_k of k colors for the tree (resp. path). Subset C_k has a complement $\overline{C_k}$ of k colors too. Both are represented by a multiple edge of G. Thus, to go through the edge corresponding to $\overline{C_k}$ the tree (resp. path) needs a color from $\overline{C_k}$, which is not in C_k.

Therefore, a spanning tree in G, or an st-path needs at least $k + 1$ colors. Since there are only $2k$ available colors, Theorem 13 follows.   □

Figure 7: A single color disjoint spanning tree while the minimum color cut is 2.

Other families of colored graphs exists, for exemple there is a family of graphs containing k^2 colors and $2k^2$ edges.

However, an upper bound on the ratio *max number of color disjoint st-paths/ MC-st-Cut* depending on the size of the colored graph may exists. Note that when the number of colors of the graph equals the MC-st-Cut value, then there is at least one st-path of each color.

Moreover, another pair of problems, MC-st-Cut and MC-Multi-Cut, has different relationships than their classical counterparts.

Proposition 14 *MC-Multi-Cut is equivalent to MC-st-Cut.*

Sketch of proof: First MC-st-Cut is a special case of MC-Multi-Cut.

Then take k copies $G_1, \ldots, G_k$ of graph G, one for each pair s_i, t_i to be disconnected in a MC-Multi-Cut instance. Merge all vertices $s_i \in G_i$ (resp. t_i) together into a single vertex s (resp. t). A colored st-cut in this new graph is trivialy a colored multicut in G.   □

8   Polynomial and "easy" special cases

In this section we present some restrictions under which colored problem are polynomial or approximable.

Proposition 15 *When each color has span 1, the MC-Path, MC-st-Cut, MC-Cut, 2Color Disjoint Paths and 2Min. Overlapping Paths problems are polynomial.*

Sketch of proof: We use the following transformation to extend a result from [2] on the MC-Path problem. Given a colored graph $G = (V, E, \mathcal{C})$, construct the graph $H = (V_H, E_H)$, where each vertex of V_H represents a color of G. There is an edge between two vertices of V_H if two edges with the corresponding colors are adjacent in G. Now add vertex s (resp. t) and edges between s (resp. t) and any vertex of H representing a color adjacent to s (resp. t) in G.

The colored problems in G reduce then to their classical counterparts st-paths, vertex-st-cut, vertex-disjoint paths in the graph H. For 2Min. Overlapping Paths, note that any two st-paths necessarily overlap on vertices which are vertex-st-cut of size one.

For MC-Cut, remark that a colored graph in which colors have span one can also be seen as an hypergraph in which the vertices spanned by a color form an hyperedge. The MC-Cut is the equivalent to finding a minimum hyperdeges cut in an hypergraph. This problem is polynomial [12]. □

The above transformation leads also to a trivial k-approximation algorithm for MC-Path, MC-st-Cut and MC-Cut when colors have span at most k.

Theorem 16 *When each color has span k, the MC-Path, MC-st-Cut, MC-Cut, 2Color Disjoint Paths and 2Min. Overlapping Paths problems can be approximated with f factor k.*

Sketch of proof: In the previous transformation, each color generates a number of vertices equals to its span, that is one for each connected component of the subgraph induced by its edges. The edges are no more set according to adjacencies between colors, but according to adjacencies between the aforesaid connected components of all colors.

Now assume that Γ is the value of a minimum vertex st-cut in the new graph. Since there are no more than k vertices corresponding to the same color, and since in the minimum color st-cut of the initial colored graph at most Γ components are cut, the number of colors of a minimum color st-cut in G is between Γ and Γ/k. Naturally the same reasoning works for MC-Path. For MC-Cut modify the hypergraph transformation like the previous one. □

The case of MC-Cut is quite particular, since it can be solved in polynomial time when the number of edges of each color is bounded by a constant k'.

Proposition 17 *When each color contains at most k' edges, MC-Cut can be solved in polynomial time.*

Sketch of proof: Let S be the value of the minimum edge cut of the colored graph without taking its colors into account. Then, the value of the optimal color cut is in $[S/k', S]$, and the associated edge cut must have size in $[S, k'S]$. In [8], Karger proved that there exist at most $|V|^{2k'}$ cuts of size in $[S, k'S]$. Moreover, they can be generated in polynomial time.

Hence, the problem can be solved by examining all the k'-approximated minimum cuts (about $|V|^{2k'}$) and counting the colors they use. □

Note that when the maximum colored degree of the graph is bounded, MC-Cut, MC-*st*-Cut and MC-Multi-Cut are also polynomial since the subsets of colors of span lower than the bound can be enumerated in polynomial time.

9   Conclusion

In this paper we have conducted a complexity and approximability study of combinatorial problems yielded by Shared Risk Resources Groups and their modelization with colored graphs.

We showed that *global* problems (MC-Cut and MC-Spanning Tree) and *local* problems, i.e. concerning two vertices such as MC-Path and MC-*st*-Cut, belong to different complexity classes (Table 1). Global problems turn out to be easier than local ones. If the case of the MC-Spanning Tree is solved ($\log n$ is the approximation factor), the complexity of MC-Cut is still a conjecture in the general case.

We proved that MC-Path and MC-*st*-Cut are not approximable within a factor $\log^k(n)$, and we can prove that those problems belong to the class III defined in particular in [6], that is they are not approximable within a factor $2^{\log^{1-\gamma} n}$ for some $\gamma > 0$. But inapproximability within a factor n^ε is still open. However the square construction of section 4 may allow to find a *booster* construction for MC-Path and MC-*st*-Cut [6] like the one used to show the inapproximability of Clique problem.

We also proved than no approximated min/max relation analogous to the classical max flow-min cut theorem hold for colored graphs. Note that this is not surprising since otherwise MC-*st*-Cut would admit an approximated certificate, and we believe that the problem does not admit any good approximation algorithm.

In the case of bounded span k, we provided an obvious k-approximation and proved that there is no k^ε-approximation.

References

[1] R. Bhandari. *Survivable Networks: Algorithms for Diverse Routing*. Kluwer Academic, 1999.

[2] P. Datta and A.K. Somani. Diverse routing for shared risk resource groups (SRRG) failures in WDM optical networks. In *IEEE BroadNets*, 2004.

[3] U. Feige. A threshold of ln n for approximating set cover. *J. of the ACM*, 45:634–652, 1998.

[4] M. Garey and D. Johnson. *Computers and Intractability: A Guide to the theory of NP-completeness*. Freeman NY, 1979.

[5] R. L. Graham, M. Grötschel, and L. Lovász, editors. *Handbook of combinatorics*, volume 1 ch.2 A. Frank. MIT Press, Cambridge, MA, USA, 1995.

[6] D. S. Hochbaum, editor. *Approximation Algorithms for NP-Hard Problems.* PWS Publishing Company, 1997.

[7] J.Q. Hu. Diverse routing in mesh optical networks. *IEEE/ACM ToN*, 2002.

[8] D.R. Karger. Global min-cuts in rnc and other ramifications of a simple mincut algorithm. In 4th *ACM-SIAM Symposium on Discrete Algorithms*, 1993.

[9] G. Li, B. Doverspike, and C. Kalmanek. Fiber span failure protection in mesh optical network. In *SPIE Opticomm*, 2001.

[10] M. Naor, L. J. Schulman, and A. Srinivasan. Splitters and near-optimal derandomization. In *FOCS '95: Proceedings of the 36th Annual Symposium on Foundations of Computer Science (FOCS'95)*, page 182, Washington, DC, USA, 1995. IEEE Computer Society.

[11] D. Papadimitriou, F. Poppe, J. Jones, S. Venkatachalam, S. Dharanikota, R. Jain, R. Hartani, and D. Griffith. Inference of shared risk link groups. IETF Draft, OIF Contribution, OIF 2001-066.

[12] R. Rizzi. On minimizing symmetric set functions. Technical report, University of Trento, 1999.

[13] P. Sebos and J. Yates et.al. Auto-discovery of shared risk link groups. In *IEEE OFC*, 2001.

[14] J. Strand. Issues for routing in the optical layers. *IEEE Com. Mag.*, 2001.

[15] V.V. Vazirani. *Approximation Algorithms.* springer, 2002.

[16] D. Xu, Y. Xiong, and C. Qiao. Protection with multi-segments (PROMISE) in networks with shared risk link groups (SRLG). *IEEE/ACM Transactions on Networking*, 11(2):248–258, 2003.

[17] S. Yuan and J.P. Jue. Dynamic path protection in WDM mesh networks under risk disjoint constraints. In *IEEE Globecom*, 2004.

[18] S. Yuan, S. Varma, and J.P. Jue. Minimum-color path problems for reliability in mesh networks. In *IEEE InfoCom*, 2005. 59-04.

A   Proofs

Theorem 16 (p. 13) *When each color has span 1, the MC-Path, MC-st-Cut, MC-Cut, 2Color Disjoint Paths and 2Min. Overlapping Paths problems are polynomial.*

Proof: A path in H correponds to a colored path in G and an st-vertex-cut in H actually represents a colored st-cut in G. Furthermore, two vertex-disjoint paths between s and t in H are trivially color disjoint in G. If no vertex-disjoint paths exist between s and t, then there is a vertex $c_1 \in H$ corresponding to a color crossed by every st-path. The set $\mathcal{C}'$ of such colors can be found by recursively computing disjoint paths from s to c_1 and c_1 to t and so on. Thus two st-paths both cross every color from $\mathcal{C}'$, and the theorem follows.

Finally for MC-Cut, remark that a colored graph which colors have span one can also be seen as an hypergraph where the minimum cut problem is polynomial [12]. Each color generates an hyperedge composed of every vertex adjacent to that color.   □

Theorem 3 (p. 6) *The MC-st-Cut problem is NP-Hard.*

Proof: [Theorem 3] In the reduction each set s_j of the Set Cover instance corresponds to a color c_j of the MC-st-Cut instance and a colored graph is constructed as follows.

Begin with two vertices s and t. For each element u_i of the Min. Set Cover instance add a path between s and t containing an edge of color c_j for all s_j containing u_i in the Min. Set Cover instance. The graph obtained has n parallel paths between s and t (Fig. 3). Therefore disconnecting s and t requires to cut each of these paths.

A cut of k colors corresponds to k sets that hits all the paths between s and t, that is all the elements of the Min. Set Cover instance, it is thus a cover. On the other hand, a cover for the Min. Set Cover instance corresponds to a set of colors of the MC-st-Cut instance, and since the cover hits every elements, the set of colors intersects all the paths of the MC-st-Cut instance. Hencefore Min. Set Cover is a special case of MC-st-Cut of the same size.   □

Theorem 12 (p. 11) *The MC-Path and MC-st-Cut problems are not approximable within a factor $o(\log |V|)$ unless $NP \subseteq TIME(n^{O(\log \log n)})$.*

Proof: [Theorem 12] Fig. 3 illustrates the reduction of Min. Set Cover to MC-Path given in [18]. The number of vertices and colors of the Minimum Color Path instance constructed as in [18] is polynomial in $|U|$ and $|S|$ of the Set Cover instance. Thus approximating the path problem within a factor $o(\log |V|)$ would mean approximating the Set Cover within a factor $o(\log |U|)$ too which is hard unless $NP \subseteq TIME(n^{O(\log \log n)})$ [3, 10].   □

Lemma 6 (p. 8) *Let G be a colored graph, a and b two vertices of G and $p \in \mathbb{N}$. Optimal solutions of MC-Path in G and in the graph G_{ab}^p of G satisfy the relation $|OPT(G_{ab}^p)| = |OPT(G)|^p$.*

Proof: [Lemma 6] We first prove that $OPT(G_{ab}^2) \leq OPT(G)^2$ and then the other way.

A Consider an optimal solution for the MC-Path in G using color set C_{opt} and remark that the Carthesian product $C_{opt} \times C_{opt}$ is a colored path in G_{ab}^2. Indeed if the path P connects a to b using only colors in C_{opt} in G, the path of G_{ab}^2 obtained by replacing each edge of P in G by the path P provides a feasible solution in G_{ab}^2 and uses only colors in $C_{opt} \times C_{opt}$.

B Consider now a solution for G_{ab}^2, and the associated path P induced on G, that is think of the copies of G in the construction of G_{ab}^2 as the edges they represent in G. In the following we refer to P as the *external path*. Let $C_P = \{c_{i_1}, c_{i_2}, \ldots, c_{i_p}\}$ the set of colors used by path P in G. If path P contains several edges of the same color c_i one may assume that inside the corresponding copies of G in G_{ab}^2 the path uses the same *internal subpath* P_i and thus the same color set $\{c_i\} \times C_{P_i}$.

Then the number of colors used by the solution on G_{ab}^2 is $|C_{P_{i_1}}| + |C_{P_{i_2}}| + \ldots + |C_{P_{i_p}}|$. Suppose w.l.o.g. that $|C_{P_{i_1}}| \leq |C_{P_{i_2}}| \leq \ldots \leq |C_{P_{i_p}}|$, therefore a better solution is to replace each internal subpath P_i with the shortest subpath P_{i_1}. The cost is decreased to $|C_{P_{i_1}}||C_P|$. Now replace the external path P with P_{i_1} if $|C_{P_{i_1}}| \leq |C_P|$, otherwise replace P_{i_1} with P.

An optimal solution for G_{ab}^2 is thus composed of optimal solutions for G : the external path has to be a shortest path in G as well as the internal paths, otherwise contradicting the optimality of solution for G_{ab}^2.

This proof generalizes to the case of $G_{a,b}^p$, there are then p levels of paths instead of only the external and internal ones. $\quad\square$

Lemma 7 (p. 8) *Let $\mathcal{A}$ be an algorithm for MC-Path or MC-st-Cut, then $\forall p \in \mathbb{N}$ a colored path P_G of G of cost lower than $|\mathcal{A}(G_{ab}^p)|^{\frac{1}{p}}$ can be deduced from $\mathcal{A}(G_{ab}^p)$.*

Proof: [Lemma 7] Algorithm $\mathcal{A}$ returns a solution for G_{ab}^p of cost C_p which can be improved in the same way as in part **B** of the proof of lemma 6. The number of colors of such a solution verifies $C_{P_G}^p \leq C_p$ where C_{P_G} is the cost of a minimum color path P_G found while decomposing solution C_p like in part **B**. Note that P_G can be a path of any level of G_{ab}^p in the initial solution of cost C_p. Anyway P_G is a minimum color path in G verifying $C_{P_G} \leq [C_{P_G}^p]^{\frac{1}{p}} \leq [C_p]^{\frac{1}{p}}$. $\quad\square$

Theorem 5 (p. 6) *For any constant $p \in \mathbb{N}^+$ the MC-Path and MC-st-Cut problems are hard to approximate within a factor $O(\log^p |V|)$ unless $NP \subseteq TIME(n^{O(\log \log n)})$.*

Proof: [Theorem 5] Suppose algorithm $\mathcal{A}$ is a $O(\log^p |V|)$-approximation algorithm for some $p \in N$ and let OPT be the value of an optimal solution. Then there exists a constant α such that for any instance (G, a, b) of MC-Path $\mathcal{A}(G) \leq \alpha(\log^p |V|)$OPT.

Now construct G_{ab}^{p+1} and run algorithm $\mathcal{A}$ on it. Thus we have $\mathcal{A}(G_{ab}^{p+1}) \leq \alpha \log^p(|V|^{p+1})OPT_{p+1}$ where OPT$_{p+1}$ is the cost of an optimal path in G_{ab}^{p+1}.

Lemma 7 tells us that a solution P_G can be derived from $\mathcal{A}(G_{ab}^{p+1})$ which cost satisfies $C_{P_G} \leq [\mathcal{A}(G_{ab}^{p+1})]^{\frac{1}{p+1}}$, and lemma 6 tells us that OPT$_{p+1}$ = OPT^{p+1}.

Therefore $C_{P_G} \leq [\alpha(p+1)^p \log^p |V|]^{\frac{1}{p+1}}.\mathrm{OPT} = \alpha^{\frac{1}{p+1}}(p+1)^{\frac{p}{p+1}} \log^{\frac{p}{p+1}} |V|.\mathrm{OPT}$.

In other words, the existence of a $O(\log^p |V|)$ approximation algorithm implies the existence of a $o(\log |V|)$ approximation algorithm for MC-Path which is not possible unless $NP \subseteq TIME(n^{O(\log \log n)})$ (theorem 12).   □

Proposition 8 (p. 9) *The MC-Path and MC-st-Cut problems are NP-Hard when the span of a color is bounded by two and are hard to approximate within a factor $1 + \varepsilon_1$ for some $\varepsilon_1 > 0$.*

Proof: [Proposition 8] We reduce the problem to Maximum 3 Satisfiability, with m clauses and n variables. For each variable x_i, $i \in \{1, \ldots, n\}$ we introduce a vertex u_i and for each clause a vertex v_j $i \in \{1, \ldots, m\}$, and then an other vertex v_{m+1}. To simplify the writing, u_{n+1} is a second name for the vertex v_1.

Then for $i \in \{1, \ldots n\}$ two disjoint paths connect u_i to u_{i+1} . The first (resp. second) path uses colors $T_{x_i,c_1}, T_{x_i,c_2}, \ldots T_{x_i,c_{l_i}}$ (resp. $F_{x_i,c_1}, F_{x_i,c_2}, \ldots F_{x_i,c_{l_i}}$) where c_j is the jth clause in which the variable x_i appears (negated or not), and l_i is the total number of clauses in which variable x_i appears.

Furthermore for each clause c_j we connect v_j to v_{j+1}, $j \in \{1, \ldots, m\}$ with three parallel edges, each edge corresponding to a literal y present in the clause c_j and which color is either F_{x_l,c_j} or T_{x_l,c_j} depending on $y = \overline{x_l}$ or $y = x_l$ for some $l \in \{1 \ldots n\}$ (see figure 5).

In the graph thus constructed the MC-Path instance of insterest is to find a path from u_1 to v_{m+1} using a minimum number of colors.

For each variable x_i we must choose a path from u_i to u_{i+1} which corresponds to a truth setting. Indeed either the set of colors $T_i = \{T_{x_i,c_1}, \ldots T_{x_i,c_{l_i}}\}$ or the set $F_i = \{F_{x_i,c_1}, \ldots F_{x_i,c_{l_i}}\}$ is used between u_i and u_{i+1}. If F_i is used for variable x_i then x_i is assigned the "false" value. Note that to cross from v_c to v_{c+1} for any clause c in which the literal $\overline{x_i}$ appears, no additional color is needed since there is an edge between v_c and v_{c+1} of a color from F_i.

Therefore when the formula is satisfiable the clause gadget can be crossed without using any additionnal color.

Consider a path connecting u_1 and v_{m+1}, it defines an assignement of the variables, and has cost $\sum_{i=1,\ldots n} l_i$ if and only if the assignement satifies all the clauses. Otherwise one extra color is needed to cross each unsatisfied clause.

Since $\sum_{i=1,\ldots,n} l_i = 3m$, the minimum cost of a path is $3m + C'$ where C' is the number of unsatisfied clauses.

In these settings, a minimum color path trivially gives an optimal solution to the Maximum 3 Satisfiability instance, and an optimal truth assignement of the variables of the Maximum 3 Satisfiability instance represents a minimum color path of the instance constructed. That proves the NP-hardness of MC-Path.

In addition, it's hard to decide if the minimum number of clauses unsatisfied is 0 or $\varepsilon_{\mathrm{sat}} m$ for some $\varepsilon_{\mathrm{sat}} > 0$ ([15]). Hence it's hard to decide if the optimal number of colors is $3m$ or $3m + \varepsilon_{\mathrm{sat}} m$, which complets the proof.

For the MC-*st*-Cut problem, the construction is similar, on the only difference is that any set of edges connected in parallel in the path construction are now connected in serie, while the egdes connected in serie are now connected in parallel. The objective is to disconnect two vertices instead of finding a path between them. □

Theorem 10 (p. 10) *The MC-Spanning Tree problem is NP-Hard even if each color has span one.*

Proof: [Theorem 10] We present a reduction to Set Cover. Again we construct a colored graph G from an instance of Set Cover where each set s_i from the Cover instance is identified with a color c_i (figure 3). The vertex set of the colored graph is the set of elements $U = \{u_1, \ldots, u_n\}$ of the set Cover instance with an additional vertex u_0.

For each set s_i there is a clique of color c_i in G involving the vertices of $s_i \cup \{u_0\}$. From the Cover point of view, it is as if u_0 belonged to every set s_i which has not the least influence upon the solutions of the cover instance while it guaranties that any cover gives a *connected* subset of colors for the spanning tree instance.

Trivialy a spanning set of colors in G is a solution of the cover instance and vice versa. □

Theorem 11 (p. 10) *The greedy algorithm for MC-Spanning Tree is a $\mathcal{H}_{|V|-1}$-approximation algorithm, where $\mathcal{H}_{|V|-1} = 1 + \frac{1}{2} + \cdots + \frac{1}{|V|-1} = O(\log |V|)$.*

Proof: [Theorem 11] Let $n = |V|$. During the algorithm $n-1$ connected components have

---

1. **Initialization :** $E \leftarrow \emptyset$, $G \leftarrow (V, E)$, $\mathcal{C}' \leftarrow \emptyset$, $i = 0$,

2. **While** G is not connected **do**

   (a) **Compute** n_c^i for all c,

   (b) **Choose the color** c_i of minimum distributed cost $w_{c_i}/n_{c_i}^i$,

   (c) $\mathcal{C}' \leftarrow \mathcal{C}' \cup \{c_i\}$, $E \leftarrow E \cup c_i$, $i \leftarrow i + 1$,

3. **Return** $\mathcal{C}'$,

Figure 8: MC-Spanning Tree approximation algorithm

to be removed. W.l.o.g. we suppose that color c_i is chosen at iteration i and that there are t iterations.

Let us look at the beginning of iteration i, we suppose that there are still n_i components to remove and that the kth removal occurs during this iteration for a cost which is thus $\alpha_k = w_{c_i}/n_{c_i}^i$.

Then since there remain exactly $(n-1)-(n-k) = n-k$ connected component to remove just before removing the kth one, $n_i \geq n - k$.

A colored tree could be obtained just by the addition of a subset of colors from an optimal colored tree for a cost lower than the cost of an optimal solution OPT. As a consequence
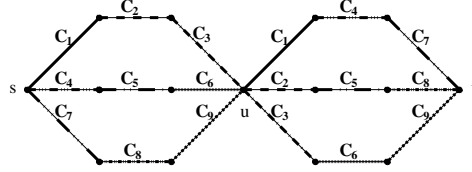
Figure 9: A single color disjoint st-path while the minimum color st-cut is 3.

the distributed cost of the color c_i chosen at iteration i, which is also the cost of the kth removal, must satisfy $\frac{w_{c_i}}{n_{c_i}^i} \leq \frac{OPT}{n_i}$, that is $\alpha_k = \frac{w_{c_i}}{n_{c_i}^i} \leq \frac{OPT}{n-k}$.

Finaly, since the cost of each chosen color has been distributed among the $n-1$ removals, the cost of the colored tree obtained is the sum of the costs of each removal : $\alpha_1 + \alpha_2 + \cdots + \alpha_{t-1} + \alpha_t \leq OPT(\frac{1}{n-1} + \frac{1}{n-2} + \cdots + \frac{1}{2} + 1) = \mathcal{H}_{n-1}OPT$. ☐

Another proof of lack of tight relation between the MC-st-Cut and Color Disjoint Paths problems.

Proof: A graph of the familly we present here contains k^2 colors and $2k^2$ edges and can be constructed as follows.

Let s, u and t be three vertices. First add k parallel color disjoint paths of k colors each between s and u. It is possible since k^2 colors are available. So far each color is represented on a single edge.

Then add k parallel disjoint color paths between u and t of length k. Each of these paths must contain a color from each su-path, as depicted in Fig. 9). Hence a su-path and a ut-path are not color disjoint.

Therefore, a path from s to t uses in its su section k colors, each of which belongs to one of the ut-paths and thus no ut-section are left for a second path to use.

☐