HAL
open science

Hardness and Approximation of Traffic Grooming

Omid Amini, Stéphane Pérennes, Ignasi Sau

To cite this version:

Omid Amini, Stéphane Pérennes, Ignasi Sau. Hardness and Approximation of Traffic Grooming. [Research Report] RR-6236, INRIA. 2007, pp.17. inria-00158341v3

HAL Id: inria-00158341

https://inria.hal.science/inria-00158341v3

Submitted on 4 Jul 2007

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.

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Hardness and Approximation of Traffic Grooming

Omid Amini — Stéphane Pérennes — Ignasi Sau

N° 6236

June 2007

Thème COM

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--6236--FR+ENG

Hardness and Approximation of Traffic Grooming

Omid Amini* †, Stéphane Pérennes *, Ignasi Sau *

Thème COM — Systèmes communicants
Projet MASCOTTE

Rapport de recherche n° 6236 — June 2007 — 17 pages

Abstract: Traffic grooming is a central problem in optical networks. It refers to pack low rate signals into higher speed streams, in order to improve bandwidth utilization and reduce network cost. In WDM networks, the most accepted criterion is to minimize the number of electronic terminations, namely the number of SONET Add-Drop Multiplexers (ADMs). In this article we focus on ring and path topologies. On the one hand, we provide the first inapproximability result for Traffic Grooming for fixed values of the grooming factor g, answering affirmatively the conjecture of Chow and Lin (*Networks, 44:194-202, 2004*). More precisely, we prove that Ring Traffic Grooming for fixed $g \geq 1$ and Path Traffic Grooming for fixed $g \geq 2$ are APX-complete. That is, they do not accept a PTAS unless P = NP. Both results rely on the fact that finding the maximum number of edge-disjoint triangles in a graph (and more generally cycles of length $2g+1$ in a graph of girth $2g+1$) is APX-complete.

On the other hand, we provide a polynomial-time approximation algorithm for Ring and Path Traffic Grooming, based on a greedy cover algorithm, with an approximation ratio independent of g. Namely, the approximation guarantee is $\mathcal{O}(n^{1/3}\log^2 n)$ for any $g \geq 1$, n being the size of the network. This is useful in practical applications, since in backbone networks the grooming factor is usually greater than the network size. As far as we know, this is the first approximation algorithm with this property. Finally, we improve this approximation ratio under some extra assumptions about the request graph.

Key-words: Approximation Algorithms, Traffic Grooming, Optical Networks, SONET ADM, Apx-hardness, PTAS, Inapproximability

This work has been partially supported by European project IST FET AEOLUS and COST 293 GRAAL.

* Projet Mascotte, CNRS/INRIA/UNSA, INRIA Sophia-Antipolis, 2004 route des Lucioles BP 93, 06902 Sophia-Antipolis Cedex, France {omid.amini,ignasi.sau,stephane.perennes}@sophia.inria.fr
† École Polytechnique, Palaiseau, France

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles, BP 93, 06902 Sophia Antipolis Cedex (France)
Téléphone : +33 4 92 38 77 77 —Télécopie : +33 4 92 38 77 65

Difficulté et approximation du problème de groupage

Résumé : Le groupage de traffic (Traffic Grooming) est un problème important dans les réseaux optiques. Il s'agit de regrouper des signaux de faible débit en signaux de plus haut débit, avec comme objectifs d'utiliser efficacement la bande-passante et réduire le coût du réseau. Dans les réseaux WDM, le critère le plus accepté est de minimiser le nombre de terminaisons électroniques, c'est à dire le nombre de "SONET Add-Drop Multiplexers" (ADMs). Dans cet article, nous étudions les topologies de l'anneau et le chemin.

D'abord, nous prouvons le premier résultat d'inapproximabilité pour Traffic Grooming quand le facteur de groupage g est fixé. Plus précisément, nous prouvons que Ring Traffic Grooming pour $g \geq 1$ fixé, et Path Traffic Grooming pour $g \geq 2$ fixé, sont APX-complets. En d'autres termes, ces problèmes n'acceptent pas de PTAS, à moins que P = NP. Cela fournit une réponse à la conjecture de Chow et Lin (*Networks, 44:194-202, 2004*). Notre méthode est basée sur le fait que trouver le nombre maximum de triangles arête-disjoints dans un graphe (et de façon plus générale, cycles de longueur $2g+1$ dans un graphe de maille $2g+1$) est APX-complet.

Nous proposons ensuite un algorithme d'approximation en temps polynomial pour Ring et Path Traffic Grooming, avec un rapport d'approximation indépendant de g. Plus précisement, le rapport d'approximation est $\mathcal{O}(n^{1/3}\log^2 n)$ pour toutes les valeurs de $g \geq 1$, où n est la taille du réseau. Ce résultat est utile dans les applications, car dans les réseaux de coeur, le facteur de groupage est généralement plus grand que la taille du réseau. Finalement, nous améliorons le rapport d'approximation sous certaines hypothèses sur le graphe des requêtes.

Mots-clés : Algorithme d'approximation, Groupage de traffic, Réseau optique, SONET ADM, Apx, PTAS, Difficulté d'approximation

1 Introduction

1.1 Background and Problem Definition

Optical wavelength division multiplexing (WDM) is today the most promising technology to accommodate the explosive growth of Internet and telecommunication traffic in wide-area, metro-area, and local-area networks. Using WDM, the potential bandwidth of 50 THz of a fiber can be divided into multiple non-overlapping wavelength or frequency channels. Since currently the commercially available optical fibers can support over a hundred frequency channels, such a channel has over one gigabit-per-second transmission speed. However, the network is usually required to support traffic connections at rates that are lower than the full wavelength capacity. In order to save equipment cost and improve network performance, it turns out to be very important to aggregate the multiple low-speed traffic connections, namely *requests*, into higher speed streams. Traffic grooming is the term used to carry out this aggregation, while optimizing the equipment cost. In WDM optical networks the most accepted criterion is to minimize the number of electronic terminations, which is unanimously considered as the dominant cost, rather than the number of wavelengths.

SONET ring is the most widely used optical network infrastructure today. In these networks, a communication between a pair of nodes is done via a *lightpath*, and each lightpath uses an Add-Drop Multiplexer (ADM), i.e. an electronic termination, at each of its two endpoints. If each request uses $\frac{1}{g}$ of the capacity of a wavelength, g is said to be the *grooming factor*. The problem is equivalent to assigning a wavelength to each request in such a way that for any wavelength and any link of the network, there can be at most g requests using this link on this wavelength. The aim is to minimize the total number of ADMs. In the graph-theoretical approach that we use, the set of requests is modeled by a graph R, and each vertex in the subgraph of R corresponding to a wavelength represents an ADM. The problem, in the case where the communication network is a ring, can be formally stated as follows:

Ring Traffic Grooming
Input: A cycle C_n on n vertices (network), a graph R (set of requests) on vertices of C_n, and a grooming factor g.
Output: Find for each edge $r = \{x, y\}$ of R, a path $P(r)$ in C_n between x and y, and a partition of the edges of R into subgraphs R_ω, $1 \leq \omega \leq W$, such that for each edge e in $E(C_n)$ and for all ω, the number of paths $P(r)$ using e, r being an edge of R_ω, is at most g.
Objective: Minimize $\sum_{\omega=1}^{W} |V(R_\omega)|$.

The statement of Path Traffic Grooming is analogous, replacing C_n by P_n.

Example 1 *Consider a ring on five nodes and the complete graph of* Fig. *1 as request graph, and let $g = 2$. We exhibit two valid solutions of the problem, both using two subgraphs (i.e. two wavelengths). The second solution is better because it uses 9 vertices instead of 10.*

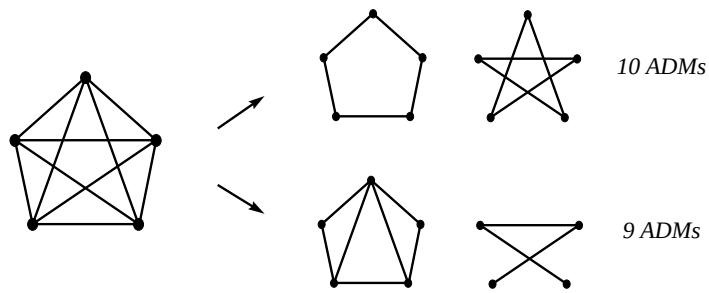

Figure 1: Two valid partitions of K_5 when $g = 2$, using different number of ADMs

1.2 Previous Work and our Contribution

The notion of traffic grooming was introduced in [GRS98] for the ring topology. Since then, traffic grooming has been widely studied in the literature (c.f. [DR02, ML01, ZM03] for some surveys). The problem has been proved to be NP-complete for ring networks and general g [CM00]. Many heuristics have been done [DR01], but exact solutions have been found only for certain values of g and for the uniform all-to-all traffic case in unidirectional ring [BC06], bidirectional ring [BCMS06], and path topologies [BC06]. On the other hand, there was no result on the inapproximability of the problem for fixed $g \geq 1$. In [CL04] the authors conjecture that Traffic Grooming is Max SNP-hard (or equivalently, APX-hard, modulo PTAS-reductions) for any fixed value of the grooming factor. We answer affirmatively to this question in Theorem 2, providing the first hardness result for the Ring Traffic Grooming problem for fixed values of the grooming factor g.

Considering g as part of the input, in [HDR06] it was proved that Path Traffic Grooming does not accept a constant-factor approximation unless P = NP. For fixed values of g, Path Traffic Grooming was proved to be in P for $g = 1$ [BC06], but the complexity for fixed $g \geq 2$ has been an open question for a while. Recently, it has been proved in [SUZ07] that Path Traffic Grooming for fixed $g > 1$ is NP-complete for *bounded number of wavelengths*. Our method permits us to improve this result in Section 3, by proving the APX-completeness of Path Traffic Grooming for any fixed $g > 1$ and unbounded number of wavelengths. In particular, this extends the NP-completeness result of [SUZ07] to the case where the number of wavelengths is not bounded.

The main ingredient of our approach is the proof of the APX-completeness (given in Section 2) of the problem of finding the maximum number of edge-disjoint triangles in a graph with bounded degree B: Maximum Bounded Edge Covering by Triangles (MECT-B for short). The proof is obtained by L-reduction from Maximum Bounded Covering by 3-Sets, which was proved to be MAX SNP-complete in [Kan91]. A simple modification of this technique permits us to prove the APX-completeness of finding the maximum number of edge-disjoint odd cycles of given length in a graph. This later claim is

then used to extend our results to arbitrary values of g, see Section 2, 3 and Appendix A, B.

The design of approximation algorithms for TRAFFIC GROOMING is the topic of the second part of this paper. We present the results for the ring topology, but the same algorithm works also for the path topology. As we show in Section 3, it is trivial to obtain a $\mathcal{O}(\sqrt{g})$-approximation with running time polynomial in g and n. For $g = 1$, the best algorithm in rings achieves an approximation ratio of 10/7 [EL04]. For general g, the best approximation algorithm [FMSZ05] achieves an approximation factor of $\mathcal{O}(\log g)$, but the problem is that the running time is exponential in g (that is, n^g). Since in practical applications SONET WDM rings are widely used as backbone optical networks [ML01, DR02], the grooming factor is usually greater than the size of the network, i.e. $g \geq n$. For those networks, the running time of this algorithm becomes exponential in n. Thus, it turns out to be important to find good approximation algorithms with running time polynomial in both n and g. In Section 4 we provide such an approximation algorithm, considering g as part of the input. Our algorithm finds a solution of RING TRAFFIC GROOMING that approximates the optimal value within a factor $\mathcal{O}(n^{1/3} \log^2 n)$ for any $g \geq 1$. To the best of our knowledge, this is the first polynomial-time approximation algorithm for the RING TRAFFIC GROOMING problem with an approximation ratio which does not depend on g. Although the performance of this algorithm seems not to be very good at first sight, in fact we conjecture that for the general instance of the problem it is not possible to get rid of a factor n^δ, for some constant $\delta > 0$. Finally, we show that the general scheme of the algorithm yields a $\mathcal{O}(\log^2 n)$-approximation if the request graph excludes a fixed graph as minor, for example if R is planar or of bounded genus. The main theoretical contribution of the second part of this paper is to relate the TRAFFIC GROOMING problem to the DENSE k-SUBGRAPH problem [FPK01]. We conclude by proposing some further research directions to better understand the complexity of TRAFFIC GROOMING.

2 APX-completeness of MECT-B

In complexity theory, the class APX (Approximable) stands for all NP-hard optimization problems that can be approximated within a constant factor. The subclass PTAS (Polynomial Time Approximation Scheme) contains the problems that can be approximated in polynomial time within a ratio $1 + \varepsilon$ for *all* constants $\varepsilon > 0$. Intuitively, these problems are the easiest ones among all NP-complete problems. Since, assuming P $\neq$ NP, there is a strict inclusion of PTAS into APX (for instance, VERTEX COVER $\in$ APX $\setminus$ PTAS), an APX-hardness result for a problem implies the non-existence of PTAS. MECT-B has been proved to be NP-complete [Hol81], and the APX-hardness when requiring node-disjoint triangles was proved in [Kan91]. The proof of the APX-hardness of MECT-B that we provide can be extended to obtain the APX-completeness of the problem of finding the maximum number of edge-disjoint cycles of length $2g + 1$ for any fixed $g \geq 1$, as sketched in Appendix A. For convenience, we prove the MAX SNP-hardness of MECT-B, which is known to be

the same as the APX-hardness modulo PTAS-reductions. MECT-B is trivially in APX, since a simple greedy algorithm provides a 3-approximation.

Theorem 1

(a) MECT-B, $B \geq 10$ *is* MAX SNP*-complete. Furthermore, the problem remains* MAX SNP*-complete in tripartite graphs.*

(b) More generally, given a $(2g+1)$*-partite graph* G *of girth* $2g+1$*, consisting of* $(2g+1)$ *parts* $A_0, \ldots, A_{2g}$ *such that the only edges are between* A_i *and* $A_{i+1} \pmod{2g+1}$*,* $i = 0, \ldots, 2g$*, and such that all the graphs induced by* $V(G) \setminus A_i$ *in* G*, for all* $i = 0, \ldots, 2g$*, form a forest, the problem of finding the maximum number of edge disjoint* C_{2g+1}*'s in* G *is* APX*-complete.*

Proof: We give here the proof of part *(a)*, the proof of part *(b)* is given in Appendix A.

L-reduction from MAX 3SC-B[1] and L-reduction to INDEP. SET-B[2]:
We define h : MECT-B $\rightarrow$ INDEP. SET - (3/2(B-2)) as follows: given a graph G as instance I of MECT-B, we define the following instance $h(I)$ of INDEP. SET - (3/2(B-2)): the graph $h(G)$ contains a node v_T for every triangle T in G. There is an edge $\{v_{T_0}, v_{T_1}\}$ in $h(G)$ iff T_0 and T_1 share an edge in G. Given a solution A of $h(I)$, we define a solution $S_h(A)$ of I by taking the triangles corresponding to nodes in A. It is easily verified that (h, S_h) is an L-reduction.

Now, we define f : MAX 3SC-B $\rightarrow$ MECT-(3B+1) in the following way: suppose that we are given as instance I, a collection C of 3-element subsets of a set X such that every element of X belongs to at most B members of C. The problem for I consists in finding the maximal number $OPT(I)$ of disjoint subsets in C. We construct an instance $f(I)$ of MECT-(3B+1), i.e. we construct a graph $G = (V, E)$ in which we ask for the maximum number $OPT(f(I))$ of edge-disjoint triangles. Let $C = \{c_1, \ldots, c_r\}$, with $|c_i| = 3$. The local replacement f substitutes for each element $c_i = \{x, y, z\} \in C$, the graph $G_i = (V_i, E_i)$ depicted in FIG. 2.

To avoid confusion, note by t any element in c_i, i.e. $t \in \{x, y, z\}$. Note that, for each element t, the nodes $t[0]$ and $t[1]$, and the edge $t[0]t[1]$ (corresponding to the thick edges in FIG. 2) appear only once in G, regardless of the number of occurrences of t. On the other hand, we add 9 new vertices $a_i[j]$, $1 \leq j \leq 9$ for each subset c_i, $1 \leq i \leq |C|$. More precisely, $G = (V, E) = \cup_{i=1}^{|C|} G_i$, where $V = \bigcup_{t \in X} \{t[0], t[1]\} \cup \bigcup_{i=1}^{|C|} \{a_i[j] : 1 \leq j \leq 9\}$ and $E = \bigcup_{i=1}^{|C|} E_i$.

Now, given a solution A of $f(I)$ of cost (or equivalently, size) c_2, we modify in polynomial time this solution to another equal or better solution A' in the following way: in each G_i, if the three triangles covering the edges $x[0]x[1]$, $y[0]y[1]$, and $z[0]z[1]$ (numbered $1, 7, 13$ in

[1] MAXIMUM BOUNDED COVERING BY 3-SETS: Given a collection of 3-subsets of a given set, each element appearing in at most B subsets, find the maximum number of disjoint subsets.

[2] MAXIMUM BOUNDED INDEPENDENT SET: Given a graph of maximum degree $\leq B$, find a maximum independent set.

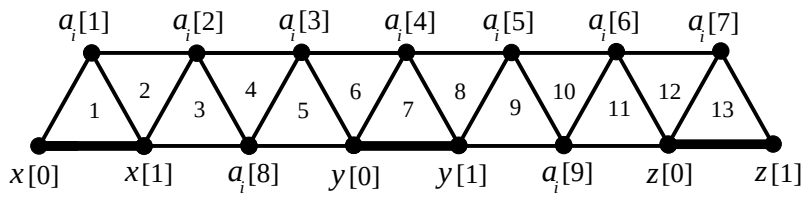

Figure 2: Gadget used in the reduction of the proof of Theorem 1

FIG. 2) belong to A, we choose the seven *odd* triangles of G_i to belong to A'. If not, we take the six *even* triangles. Let $c_2' \geq c_2$ be the cost of A'. Then, we define a solution $S_f(A)$ of I by choosing the subset c_i to be in $S_f(A)$ if and only if A' contains exactly 7 triangles in G_i. We claim that the pair (f, S_f) is an L-reduction: in each G_i there are 13 different triangles, numbered from 1 to 13 in FIG. 2. The only way to choose 7 edge-disjoint triangles in G_i is by taking all the *odd* triangles, and thus by covering the three edges $x[0]x[1]$, $y[0]y[1]$, and $z[0]z[1]$. All other choices of triangles yield at most 6 edge-disjoint triangles. The key observation is that we are able to choose 7 triangles exactly $OPT(I)$ times. Indeed, each time we choose 7 triangles we cover the edges corresponding to 3 elements of c_i, and since the number of disjoint c_i's in C is $OPT(I)$, this can be done exactly $OPT(I)$ times. On the other hand, one can easily see that $OPT(I) \geq \frac{|C|}{3B}$. Hence:

$$OPT(f(I)) = 7 \cdot OPT(I) + 6(|C| - OPT(I)) \leq OPT(I) + 18B \cdot OPT(I) = (18B+1)OPT(I)$$

To conclude, note that if the solution $S_f(A)$ of I has cost c_1, we have $OPT(I) - c_1 \leq OPT(f(I)) - c_2$. To see this, we observe that $OPT(f(I)) = 6r + OPT(I)$, and also $c_2' = 6r + c_1$, and so $OPT(f(I)) - OPT(I) = c_1 - c_2' \leq c_1 - c_2$.

Both (f, S_f) and (h, S_h) are L-reductions and MAX 3SC-B, $B \geq 3$ and INDEP. SET-B, $B \geq 5$ are MAX SNP-complete [Kan91]. Thus, MECT-B, $B \geq 10$ is MAX SNP-complete.

To prove the last claim, note that the graph $G = (V, E)$ used in the proof is a tripartite graph. Indeed, it is easy to check that the three vertex sets that define the tripartition are:

$$V_0 = \bigcup_{t \in X}^{|X|} t[0] \ \cup \ \bigcup_{i=1}^{|C|} \{a_i[2], a_i[5]\}, \qquad V_1 = \bigcup_{i=1}^{|C|} \{a_i[1], a_i[4], a_i[7], a_i[8], a_i[9]\}, \text{ and}$$

$$V_2 = \bigcup_{i=1}^{|X|} t[1] \ \cup \ \bigcup_{t \in X}^{|C|} \{a_i[3], a_i[6]\}.$$

□

3 APX-completeness of Traffic Grooming

In this section we prove the hardness results for Ring Traffic Grooming and Path Traffic Grooming. First we prove that Ring Traffic Grooming belongs to APX when g is fixed. The same result holds for Path Traffic Grooming.

Let us define the *density* ρ of a graph G as its edges-to-vertices ratio: $\rho(G) = \frac{|E(G)|}{|V(G)|}$. To see that Ring Traffic Grooming is in APX for any fixed $g \geq 1$, we have to find a constant-factor approximation algorithm. We use the fact that the best possible density ρ^* of any subgraph used in the partition of the request graph is $\mathcal{O}(\sqrt{g})$, given by (possibly a subgraph of) a circulant graph [BCMS06]. We prove that the cost A of any solution $R_1, \ldots, R_W$ is in the interval $[\frac{|E(R)|}{\rho^*}, 2|E(R)|]$. This clearly implies that any solution has cost at most $2\rho^* = \mathcal{O}(\sqrt{g})$ times the optimal cost. To see this, note that each edge of R contributes at most twice to the cost, so $A \leq 2|E(R)|$. On the other hand, we have

$$A = \sum_{\omega=1}^{W} |V(R_\omega)| = \sum_{\omega=1}^{W} \frac{|E(R_\omega)|}{\rho(R_\omega)} \geq \sum_{\omega=1}^{W} \frac{|E(R_\omega)|}{\rho^*} = \frac{|E(R)|}{\rho^*}$$

Thus, a $\mathcal{O}(\sqrt{g})$-approximation is obtained just by taking *any* partition of the request graph.

Theorem 2 Ring Traffic Grooming *is* APX*-complete for all fixed* $g \geq 1$. *Thus, it does not accept a* PTAS *unless* P = NP.

Proof: Sketch. We prove that Ring Traffic Grooming is APX-complete even if we suppose that the degree of the request graph is bounded by a constant $B \geq 10$. First, we prove the result for $g = 1$. We consider a set of requests R made of a tripartite graph with the three partition classes placed consecutively on the ring, as shown in Fig. 3a. To simplify the presentation, suppose that R can be partitioned into triangles. In any solution, the only possible involved subgraphs are P_2, P_3, P_4, and K_3. It is clear that the best we can do is to groom the requests into triangles (since triangles have the highest density) obtaining an optimal cost of $|E(R)|$. From this we derive that $|E(R)|$ is a lower bound for the number of ADMs of any solution, and that each path used in a given solution adds an additional unity of cost. For each solution S, the additional cost is at least 4/3 times the number of edges covered by paths of S. This bound is tight if all the paths are P_4's. Thus, the number A of ADMs used by S (i.e. the cost of S), satisfies $A \geq (1-\varepsilon)|E(R)| + \varepsilon\frac{4}{3}|E(R)| = (1+\frac{\varepsilon}{3})|E(R)|$, where ε is the percentage of edges of R not covered by triangles in S. By Theorem 2, there exists a constant ε_0 such that we can find in polynomial time at most a fraction $(1-\varepsilon_0)$ of the triangles of R. This means that $(1+\frac{\varepsilon_0}{3})OPT$ is the best solution we may obtain by a polynomial-time algorithm, implying the non-existence of a PTAS.

For $g > 1$, we take a $(2g+1)$-partite graph as the request graph, in such way that each cycle makes at least g tours around the center of the ring. Now, we can reduce the grooming problem to the problem of finding a maximum number of cycles of length $2g+1$ in this graph (as in the case $g = 1$). This later problem is also APX-complete, see Theorem 1 and Appendix A. The details can be found in Appendix B. Hence, Ring Traffic Grooming is Max SNP-complete for bounded number of requests per node $B \geq 10$. □

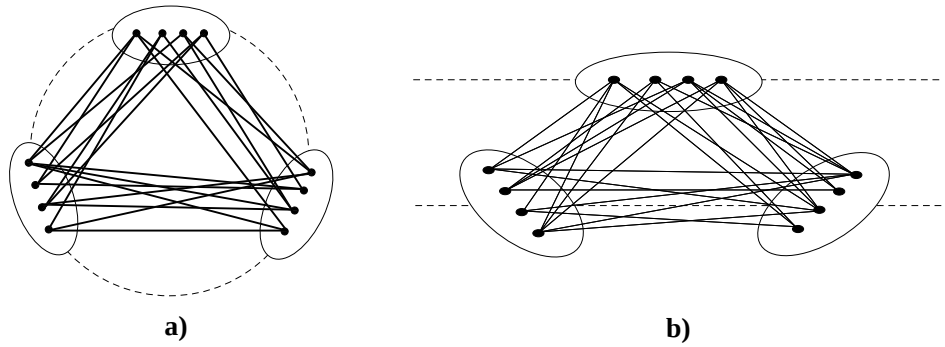

Figure 3: Request graph used in the proof of APX-completeness of TRAFFIC GROOMING: **a)** in the ring for $g = 1$; **b)** in the path for $g = 2$

Theorem 3 PATH TRAFFIC GROOMING *is* APX*-complete for any fixed* $g \geq 2$.

Proof: Again, the result holds even for bounded number B of requests per node, $B \geq 10$. We prove the result for $g = 2$, proceeding for $g > 2$ as in the proof of Theorem 2. Consider a set of requests R made of a tripartite graph with the three partition classes placed consecutively on the path one after another, as shown in FIG. 3b. Since each triangle induces load 2, minimizing the number of ADMs corresponds to finding the maximum number of edge-disjoint triangles. Therefore, it does not accept a PTAS unless P = NP. □

4 Approximating Ring Traffic Grooming

We are now interested in finding good approximation algorithms considering g as part of the input. As we saw in Section 3, obtaining a $\mathcal{O}(\sqrt{g})$-approximation is trivial. Since in practical applications SONET WDM rings are widely used as backbone optical networks [ML01, DR02], the grooming factor is usually greater than the size of the network, i.e. $g \geq n$. Thus, it turns out to be important to find approximation algorithms with an approximation ratio not depending on g. A general approximation algorithm with this property is the main result of this section. It provides in the worst case a $\mathcal{O}(n^{1/3} \log^2 n)$-approximation. We describe it for the ring, but exactly the same arguments provide an algorithm for the path. The main idea is to greedily find subgraphs with high density using approximation algorithms for the DENSE k-SUBGRAPH problem, which is defined as follows: given a graph G and an integer k, find an induced subgraph $H \subseteq G$ on k vertices with the highest density among all subgraphs on k vertices. In [FPK01] the authors provide a polynomial-time algorithm with approximation ratio $2n^{1/3}$. To simplify the presentation, suppose that $n = 2^t$ for some $t > 0$:

Algorithm $\mathcal{A}$:

Step **1)** Divide the request set into $\log n$ classes, such that in each class C_i the length of the requests lies in the interval $[2^i, 2^{i+1})$, $i = 0, \ldots, \log n - 1$. For each class C_i, the ring can be divided into intervals of length 2^i such that the only requests are between consecutive intervals. In this way we obtain $\frac{n}{2^i}$ subproblems for each class: each one consists in finding an optimal solution in a bipartite graph of size $2 \cdot 2^i$. More precisely, each subproblem can be formulated as:

BIPARTITE TRAFFIC GROOMING
Input: A bipartite graph R, and a grooming factor g.
Output: Partition of the edges of R into subgraphs R_ω with at most g edges, $1 \leq \omega \leq W$.
Objective: Minimize $\sum_{\omega=1}^{W} |V(R_\omega)|$.

Solve all these BIPARTITE TRAFFIC GROOMING subproblems independently, and output the union of all solutions.

Step **2)** To solve each BIPARTITE TRAFFIC GROOMING subproblem in a bipartite graph R, proceed greedily (until all edges are covered) by finding at step i a subgraph R_i of $G \setminus (R_1 \cup \cdots \cup R_{i-1})$ with at most g edges in the following way:
For each $k = 2, \ldots, 2g$ find a subgraph B_k of $R \setminus (R_1 \cup \cdots \cup R_{i-1})$ using the algorithm of [FPK01] for the DENSE k-SUBGRAPH problem.

- If for some k^*, $|E(B_{k^*})| > g$, and $|E(B_i)| \leq g$ for all $i < k^*$, remove $|E(B_k^*)| - g$ arbitrary edges of B_{k^*} and replace B_k^* with this new graph. Stop the search at k^*, and output the densest graph among $B_2, \ldots, B_{k^*-1}, B_{k^*}$.
- If not, output the densest subgraph among $B_2, \ldots, B_{2g}$.

Let OPT be the optimal solution of RING TRAFFIC GROOMING, and let OPT_1 be the cost of the solution obtained by solving *optimally* all the subproblems generated by Step **1** of Algorithm $\mathcal{A}$. We prove a lemma before stating the theorem.

Lemma 1 *Let β be a given number. Suppose that we can find in any bipartite graph R on at most n vertices, a subgraph with at most g edges which has density at least $1/\beta$ times the density of the densest subgraph with at most g edges. Then in the greedy procedure of Step 2 of Algorithm $\mathcal{A}$ we obtain a solution of cost OPT_2 such that $OPT_2 \leq \mathcal{O}(\log n) \cdot \beta \cdot OPT_1$.*

Proof: Let m be the number of edges of the request graph R, and let $R_1, R_2, \ldots, R_r$ be the subgraphs generated in order by the above algorithm which cover all the edges, i.e. R_1 is

chosen first, R_2 is chosen second, and so on. We will prove

$$\sum |V(R_i)| \leq \log(m) \cdot \beta \cdot OPT_1$$

To prove this, we first enumerate the edges of R in order of appearance in R_i's: all the edges in R_1 will be enumerated $e_1, \ldots, e_{g_1}$ ($g_1 = |E(R_1)| \leq g$), all the edges in R_2 will be enumerated $e_{g_1+1}, \ldots, e_{g_1+g_2}$ ($g_2 = |E(R_2)| \leq g$), and so on. Let ρ_i be the density of the subgraph R_i, i.e. $\rho_i = \frac{|E(R_i)|}{|V(R_i)|}$, and $\Sigma = \sum |V(R_i)|$ the total cost of the solution. For every edge $e_j \in R_i$, we define $c(e_j) = \frac{1}{\rho_i}$. We claim that $\sum_j c(e_j) = \Sigma$. To prove this equality just note that $\sum_{e_j \in E(R_i)} c(e_j) = \frac{|E(R_i)|}{\rho_i} = |V(R_i)|$, and so $\sum_j c(e_j) = \sum_i |V(R_i)| = \Sigma$. Let us define R'_i to be the union of $R_i, R_{i+1}, \ldots, R_r$. We define ρ'_i to be the density of the densest subgraph of R'_i containing at most g edges. Let us take an optimal solution for R'_i, i.e. a decomposition of R'_i into subgraphs $A_1, \ldots, A_s$ such that $\sum_{k=1}^{s} |V(A_k)|$ is minimum. Let $\overline{\rho}_1, \ldots, \overline{\rho}_s$ be the density of these subgraphs. We have:

- $\forall k \leq s,\ \ \overline{\rho}_k = dens(A_k) \leq \rho'_i$: because each A_k is a subgraph of R'_i containing at most g edges, and ρ'_i is the density of the densest subgraph with at most g edges in R'_i.
- $\rho'_i \leq \beta\rho_i$: because we suppose that we can find an approximation of ρ'_i up to a factor $1/\beta$.

This implies in particular that

$$\frac{1}{\overline{\rho}_k} \geq \frac{1}{\beta\rho_i}, \quad \text{and so}$$

$$\sum_k |V(A_k)| = \sum_k \frac{|E(A_k)|}{\overline{\rho}_k} \geq \sum_k \frac{|E(A_k)|}{\beta\rho_i} = \frac{|E(R'_i)|}{\beta\rho_i}$$

But an optimal solution for R provides a solution for R'_i of cost at least the optimal solution for R'_i, i.e. $\sum_k |V(A_k)| \leq OPT_1$. Using this in the above inequality we get $\frac{1}{\rho_i} \leq \frac{\beta \cdot OPT_1}{|E(R'_i)|}$, and so for an edge $e_j \in R_i$ we have

$$c(e_j) = \frac{1}{\rho_i} \leq \frac{\beta \cdot OPT_1}{|E(R'_i)|} \leq \frac{\beta \cdot OPT_1}{m - j + 1}, \quad \text{and this proves}$$

$$\Sigma = \sum_j c(e_j) \leq \beta \cdot (\sum_j \frac{1}{m-j+1}) \cdot OPT_1 \leq \beta \cdot \log(m) \cdot OPT_1 \leq 2\beta \cdot \log(n) \cdot OPT_1$$

□

Theorem 4 *Algorithm $\mathcal{A}$ is a polynomial-time approximation algorithm that approximates* Ring Traffic Grooming *within a factor $\mathcal{O}(n^{1/3} \log^2 n)$ for any $g \geq 1$.*

Proof: Algorithm $\mathcal{A}$ returns a valid solution of RING TRAFFIC GROOMING, because each request is contained in some bipartite graph, and no request is counted twice. The running time is polynomial in both n and g, because we run at most $2g-1$ times the algorithm of [FPK01] for each subproblem, and there are $n(\sum_{i=0}^{t-1}\frac{1}{2^i})-1=2n-3$ subproblems. We prove now the approximation guarantee:

- We claim that $OPT_1 \leq 2\log n \cdot OPT$. Indeed, let $\overline{c}_i$ be the optimal cost of the subset of requests of length in the interval $[2^i, 2^{i+1})$, $i=0,\ldots,\log(n)-1$. It is clear that $\overline{c}_i \leq OPT$ for each i , and thus $\sum_{i=0}^{\log n-1}\overline{c}_i \leq \log n \cdot OPT$. Finally, $OPT_1 \leq 2\sum_{i=0}^{\log n-1}\overline{c}_i$, because each vertex is taken into account in two subproblems.

- The greedy procedure described in Step **2** of Algorithm $\mathcal{A}$ outputs a graph whose density is at least $\frac{1}{2n^{1/3}}$ times the highest density (with at most g edges) of the updated request graph. To see that, note that the optimal density is achieved by a subgraph on at most $2g$ vertices (it would be the case of g disjoint edges). Then, for each value of k, the algorithm of [FPK01] finds a $2n^{1/3}$-approximation of the maximum number of edges of an induced subgraph on k vertices[3]. Thus, if we take the densest subgraph among $B_2,\ldots,B_{2g}$ (removing edges if necessary) we also obtain a $2n^{1/3}$-approximation of the highest density of a subgraph with at most g edges. Let ρ_k be the density of B_k before removing edges. The explicit formula of the highest density ρ that we output in Step **2** of Algorithm $\mathcal{A}$ is:

$$\rho := \max_{k\in\{2,\ldots,2g\}} \min\left(\rho_k, \frac{g}{k}\right)$$

  Looking at the formula we understand why we stop at $k=k^*$ in the algorithm. In other words, we have proved that we can use $\beta = 2n^{1/3}$ in Lemma 1.

- By combining the remarks above and Lemma 1 we obtain that the cost A returned by Algorithm $\mathcal{A}$ satisfies $A \leq 2n^{1/3}\cdot OPT_2 \leq 4n^{1/3}\log n \cdot OPT_1 \leq 8n^{1/3}\log^2 n \cdot OPT$.

□

We can improve the approximation ratio of the algorithm if all the requests have short length compared to the length of the ring. This situation is usual in practical applications since nodes may want to communicate only with their nearest neighbors. Let $f(n)$ be any function of n. If all the requests have length at most $f(n)$, then the above algorithm provides an approximation ratio of $\mathcal{O}(f(n)^{1/3}\log^2 n)$. Indeed, in Step **2** of Algorithm $\mathcal{A}$, we have to find dense subgraphs in bipartite graphs of size at most $2f(n)$, hence the factor $2n^{1/3}$ can be replaced with $2(2f(n))^{1/3}$.

Remark that all the instances of DENSE k-SUBGRAPH problem in our algorithm are bipartite. Using the results of [ST05], it is possible to obtain a better approximation ratio when the request graph is bipartite and satisfies some uniform density conditions. We omit the proof due to lack of space.

[3] In fact, the improved approximation ratio of the DENSE k-SUBGRAPH problem is $\mathcal{O}(n^\delta)$ for some constant $\delta < 1/3$ [FPK01]. Obviously, the same applies to our algorithm, replacing the exponent $1/3$ with $\delta < 1/3$.

Corollary 1 *If the request graph R is such that in any large enough subgraph $H \subseteq R$, a densest subgraph $(A \cup B, E)$ satisfies $|A|, |B| = \mathcal{O}(\sqrt{g})$ and $|E| = \Omega(g)$, then for any constant $\varepsilon > 0$ there exists a polynomial-time algorithm for* Ring Traffic Grooming *with approximation ratio $\mathcal{O}(n^{\varepsilon} \log^2 n)$.*

To end this section, note that the results of [DHK05] show that the density can be approximated within a constant factor two in the class of graphs excluding a fixed graph H as minor. Thus, if the request graph R is H-minor free (for instance if R is planar, or of bounded genus,...), our algorithm achieves an approximation factor of $\mathcal{O}(\log^2 n)$.

5 Conclusions and Further Research

This article deals with hardness and approximability of the Traffic Grooming problem, a central problem in WDM optical networks. The contribution of this work can be divided in two main parts: On the one hand, we state the first hardness results for Ring Traffic Grooming and Path Traffic Grooming for fixed values of g. More precisely, we prove that Ring Traffic Grooming is APX-complete for fixed $g \geq 1$, and that Path Traffic Grooming is APX-complete for fixed $g \geq 2$. In other works, we rule out the existence of a PTAS for fixed values of g. To prove this results we reduce Ring Traffic Grooming for $g = 1$ to the problem of finding the maximum number of edge-disjoint triangles in a graph of degree bounded by B (MECT-B for short). We prove that MECT-B is APX-complete, and we generalize this reduction for Path Traffic Grooming and for all values of $g \geq 1$. On the other hand, we provide the first polynomial-time approximation algorithm for Ring and Path Traffic Grooming with an approximation ratio not depending on g, considering g as part of the input.

There remains still a lot of work to be done. It is a challenging open problem to close the complexity gap of Traffic Grooming, that is, to provide an approximation algorithm with an approximation ratio matching the corresponding inapproximability result. We are convinced that the inherent difficulty of the problem resides in finding dense subgraphs with bounded number of edges. This problem is strongly related to the problem of finding the densest subgraph with bounded number of vertices, which has been recently proved to have, essentially, the same difficulty as the Dense k-Subgraph problem [And07]. The non-existence of a PTAS for the Dense k-Subgraph problem has been proved in [Kho04] involving very technical proofs, and this is the best existing hardness result. A long-standing conjecture claims that there exists some constant $\varepsilon > 0$ such that finding a n^{ε}-approximation for Dense k-Subgraph is NP-hard [FPK01]. As we proved in Section 4, an α-approximation for Dense k-Subgraph yields a $\mathcal{O}(\alpha \log^2 n)$-approximation for Ring Traffic Grooming. We suspect that a similar result in the other direction should also exist. Because of this, we conjecture the following

Conjecture 1 *There exists some constant $\delta > 0$, such that* Ring Traffic Grooming, *when g is part of the input, is hard to approximate within a factor n^{δ}.*

Acknowledgement. Many thanks to David Coudert, Mordechai Shalom and Shmuel Zaks for helpful discussions.

References

[And07] R. Andersen. Finding large and small dense subgraphs. Submitted for publication (arXiv:cs/0702032v1), February 2007.

[BC06] J-C. Bermond and D. Coudert. *The CRC Handbook of Combinatorial Designs (2nd edition)*, chapter VI.27, Grooming. CRC Press, C.J. Colbourn and J.H. Dinitz edition, 2006. To appear.

[BCMS06] J-C. Bermond, D. Coudert, X. Muñoz, and I. Sau. Traffic Grooming in Bidirectional WDM Ring Networks. In *IEEE-LEOS ICTON / COST 293 GRAAL*, volume 3, pages 19–22, 2006.

[CL04] T. Chow and P.J. Lin. The ring grooming problem. *Networks*, 44:194–202, 2004.

[CM00] A. Chiu and E.H. Modiano. Traffic grooming algorithms for reducing electronic multiplexing costs in WDM ring networks. *IEEE/OSA Journal of Lightwave Technology*, 18(1):2–12, 2000.

[DHK05] E. Demaine, M. T. Hajiaghayi, and K. C. Kawarabayashi. Algorithmic graph minor theory: Decomposition, approximation and coloring. In *46th Annual IEEE Symposium on Fondations of Computer Science (FOCS)*, pages 637–646, October 2005.

[DR01] R. Dutta and G.N Rouskas. On Optimal Traffic Grooming in WDM Rings. In *Proceedings of ACM Sigmetrics/Performance*, pages 164–174, Cambridge, 2001.

[DR02] R. Dutta and N. Rouskas. Traffic grooming in WDM networks: Past and future. *IEEE Network*, 16(6):46–56, 2002.

[EL04] L. Epstein and A. Levin. Better Bounds for Minimizing SONET ADMs. In *WAOA*, pages 281–294, Bergen, Norway, 2004.

[FMSZ05] M. Flammini, L. Moscardelli, M. Shalom, and S. Zaks. Approximating the traffic grooming problem. In *the 16th International Symposium on Algorithms and Computation (ISAAC)*, Lecture Notes in Computer Science 3827, Springer-Verlag, pages 915–924, December 2005.

[FPK01] U. Feige, D. Peleg, and G. Kortsarz. The Dense k-Subgraph Problem. *Algorithmica*, 29(3):410–421, 2001.

[GRS98] O. Gerstel, R. Ramaswami, and G. Sasaki. Cost Effective Traffic Grooming in WDM Rings. In *Proceedings of INFOCOM*, pages 69–77, San Francisco, 1998.

[HDR06] S. Huang, R. Dutta, and G.N. Rouskas. Traffic Grooming in Path, Star, and Tree Networks: Complexity, Bounds, and Algorithms. *IEEE Journal on Selected Areas in Communications*, 24(4):66–82, April 2006.

[Hol81] I. Holyer. The NP-Completeness of Some Edge-Partition Problems. *SIAM Journal on Computing*, 10(4):713–717, 1981.

[Kan91] V. Kann. Maximum bounded 3-dimensional matching is MAX SNP-complete. *Information Processing Letters*, 37:27–35, 1991.

[Kho04] S. Khot. Ruling out PTAS for graph min-bisection, densest subgraph and bipartite clique. In *FOCS*, pages 136–145, 2004.

[ML01] E. Modiano and P. Lin. Traffic grooming in WDM networks. *IEEE Communications Magazine*, 39(7):124–129, 2001.

[ST05] A. Suzuki and T. Tokuyama. Dense subgraph problem revisited. In *Joint Workshop "New Horizons in Computing" and "Statistical Mechanical Approach to Probabilistic Information Processing"*, Japan, July 2005.

[SUZ07] M. Shalom, W. Unger, and S. Zaks. On the Complexity of the Traffic Grooming Problem in Optical Networks. In *Fun with Algorithms, 4th International Conference*, pages 262–271, 2007.

[ZM03] K. Zhu and B. Mukherjee. A review of traffic grooming in WDM optical networks: Architectures and challenges. *Optical Networks*, 4(2):55–64, 2003.

A Sketch of the proof of the APX-completeness of finding the maximum number of odd cycles

We provide here the proof of the fact that finding the maximum number of C_{2g+1}'s in a graph G is also APX-complete for any fixed $g > 1$. In fact, we need and prove a stronger result (part *(b)* of Theorem 1):

Theorem 5 *Given a $(2g+1)$-partite graph G of girth $2g+1$, consisting of $(2g+1)$ parts $A_0, \ldots, A_{2g}$ such that the only edges are between A_i and $A_{i+1} \pmod{2g+1}$, $i = 0, \ldots, 2g$, and such that all the graphs induced by $V(G) \setminus A_i$ in G, for all $i = 0, \ldots, 2g$, form a forest, the problem of finding the maximum number of edge disjoint C_{2g+1}'s in G is* APX*-complete.*

Proof: First, note that a greedy algorithm provides a constant factor approximation with factor $2g+1$. Now, consider the gadget of the proof of Theorem 1 (see Fig. 2). We modify this gadget in such a way that the same proof holds for C_{2g+1}'s instead of C_3's (triangles), and such that all the conditions of Theorem 5 are verified. Given $g > 1$, we add a chain of $4g+1$ triangles between any two pair of triangles corresponding to *thick* edges (that is, between the edges corresponding to elements of X). Then we add $g-1$ inner points to all the edges going from up to down in the triangles. An example if shown in Fig. 4.

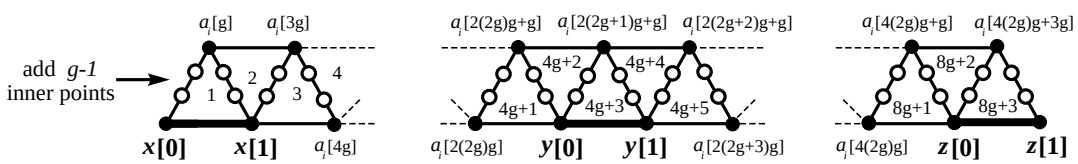

Figure 4: Adding $g-1$ inner points to prove the APX-completeness of finding edge-disjoint C_{2g+1}'s

It is easily seen that the graph built in this way is $(2g+1)$-partite. Indeed, it admits a partition into $(2g+1)$ parts, numbered $0, \ldots, 2g$, which consist of enumerating the vertices cyclically. Let $A_0, \ldots, A_{2g}$ be the different parts. In such a $(2g+1)$-partition, for any element $t \in X$, the vertex $t[0]$ belongs to A_0, and the vertex $t[1]$ belongs to A_{2g}. We need this property to ensure the consistency of our gadget when an element appears in more that one subset. Note that the graph induced by $V(G) \setminus A_i$ in G, for all $i = 0, \ldots, 2g$, form a forest. At this point, one can rewrite the proof of Theorem 1 to obtain the result, just by changing the multiplicative constants. $\square$

B Sketch of the proof of the APX-completeness of RING TRAFFIC GROOMING for $g > 1$

We generalize the proof of the case $g = 1$. Let G be a graph satisfying the conditions of Theorem 5: G is a $(2g+1)$-partite graph, consisting of $2g+1$ parts $A_0, \ldots, A_{2g}$ such that the only edges are between A_i and $A_{i+1} \pmod{2g+1}$, $i = 0, \ldots, 2g$, and such that the graph induced between two consecutive parts of G forms a forest (or more generally a graph of girth at least $g+1$). Again, in order to simplify the presentation suppose that this graph can be partitioned into C_{2g+1}'s.

Now, let $c_0, \ldots, c_{2g}$ be a permutation of the vertices of the cycle C_{2g+1}, such that the polygon $(c_0, \ldots, c_{2g})$ makes g tours around the center (for $g = 1$ take the triangle. For g arbitrary, let $c_i = exp(\frac{2ig\pi}{2g+1})$). Now replace each vertex c_i with an interval consisting of vertices of A_i. In this cyclic representation of the graph G, each cycle makes at least g tours around the origin. To see this, recall that the only possible edges are between A_i and $A_{i+1} \pmod{2g+1}$, $i = 0, \ldots, 2g$, and also the graph induced between two consecutive parts forms a forest. This implies that every cycle should intersect each A_i at least once, and so this cycle makes at least g tours around the origin, as the original cycle $\{c_0, \ldots, c_{2g}\}$ does so. Each cycle used in the solution should be of length exactly $2g+1$, there is no cycle of smaller length, and longer cycles use each edge more than g times, as they make more than g tours around the origin. Then the problem is reduced to finding edge-disjoint cycles of length $2g+1$, which is also MAX SNP-hard by Theorem 5.

The proof of Theorem 2 can now be reproduced to obtain the same result for any g, replacing the factor $\frac{4}{3}$ for $g = 1$ (because the path with highest density in any solution for $g = 1$ is a P_4) with a factor $\frac{2g+2}{2g+1}$ for a general g (because the path with highest density in any solution for general g is a P_{2g+2}).

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399