

Hardness of Approximating the Traffic Grooming

Omid Amini *Stéphane Pérennes* **Ignasi Sau**

Projet MASCOTTE, CNRS/I3S-INRIA-UNSA, France

AlgoTel'07

Plan de l'exposé

- Introduction
- Groupage de trafic
 - ▶ Définition du problème
 - ▶ Exemple
 - ▶ État de l'art
 - ▶ Problème de MECT-B
 - ▶ Résultats d'inapproximabilité
- Trouver un sous-graphe dense
 - ▶ Définition du problème
 - ▶ Relation avec le groupage
 - ▶ Résultats d'inapproximabilité
- Conclusions

Comment prouver l'inapproximabilité

- **Classe APX (Approximable):**

un problème d'optimisation est dans APX s'il admet un algorithme d'approximation avec un rapport constant.

Exemple: VERTEX COVER

- **Classe PTAS (Polynomial-Time Approximation Scheme):**

un problème d'optimisation est dans PTAS s'il admet des algorithmes d'approximation avec un rapport $1 + \epsilon$, pour tout $\epsilon > 0$ (le meilleur qu'on peut espérer pour un problème d'optimisation).

Ex.: TRAVELING SALESMAN PROBLEM *dans le plan euclidien*

- On sait que

$$\text{PTAS} \subsetneq \text{APX}$$

- Et donc, si Π est un problème d'optimisation:

$$\Pi \text{ est APX-dur} \Rightarrow \Pi \notin \text{PTAS}$$

Comment prouver l'inapproximabilité

- **Classe APX (Approximable):**

un problème d'optimisation est dans APX s'il admet un algorithme d'approximation avec un rapport constant.

Exemple: VERTEX COVER

- **Classe PTAS (Polynomial-Time Approximation Scheme):**

un problème d'optimisation est dans PTAS s'il admet des algorithmes d'approximation avec un rapport $1 + \varepsilon$, pour tout $\varepsilon > 0$ (le meilleur qu'on peut espérer pour un problème d'optimisation).

Ex.: TRAVELING SALESMAN PROBLEM *dans le plan euclidien*

- On sait que

$$\text{PTAS} \subsetneq \text{APX}$$

- Et donc, si Π est un problème d'optimisation:

$$\Pi \text{ est APX-dur} \Rightarrow \Pi \notin \text{PTAS}$$

Comment prouver l'inapproximabilité

- **Classe APX (Approximable):**

un problème d'optimisation est dans APX s'il admet un algorithme d'approximation avec un rapport constant.

Exemple: VERTEX COVER

- **Classe PTAS (Polynomial-Time Approximation Scheme):**

un problème d'optimisation est dans PTAS s'il admet des algorithmes d'approximation avec un rapport $1 + \varepsilon$, pour tout $\varepsilon > 0$ (le meilleur qu'on peut espérer pour un problème d'optimisation).

Ex.: TRAVELING SALESMAN PROBLEM *dans le plan euclidien*

- On sait que

$$\text{PTAS} \subsetneq \text{APX}$$

- Et donc, si Π est un problème d'optimisation:

$$\Pi \text{ est APX-dur} \Rightarrow \Pi \notin \text{PTAS}$$

Groupage de trafic

Introduction

- Réseaux WDM (Wavelength Division Multiplexing)

- ▶ 1 longueur d'onde (ou fréquence) = jusqu'à 40 Gb/s
- ▶ 1 fibre = centaines de longueurs d'onde = Tb/s

- Idée

Le **groupage de trafic** dans les réseaux consiste à regrouper trafic de faible vitesse dans un flux de trafic de plus haute vitesse

→ on affecte plusieurs requêtes de faible vitesse à la même longueur d'onde (TDM, Time Division Multiplexing)

- Objectifs

- ▶ Partage efficace de la bande passante
- ▶ Réduire le coût des équipements de routage

Introduction

- Réseaux WDM (Wavelength Division Multiplexing)

- ▶ 1 longueur d'onde (ou fréquence) = jusqu'à 40 Gb/s
- ▶ 1 fibre = centaines de longueurs d'onde = Tb/s

- Idée

Le **groupage de trafic** dans les réseaux consiste à regrouper trafic de faible vitesse dans un flux de trafic de plus haute vitesse

→ on affecte plusieurs requêtes de faible vitesse à la même longueur d'onde (TDM, Time Division Multiplexing)

- Objectifs

- ▶ Partage efficace de la bande passante
- ▶ Réduire le coût des équipements de routage

Introduction

- Réseaux WDM (Wavelength Division Multiplexing)

- ▶ 1 longueur d'onde (ou fréquence) = jusqu'à 40 Gb/s
- ▶ 1 fibre = centaines de longueurs d'onde = Tb/s

- Idée

Le **groupage de trafic** dans les réseaux consiste à regrouper trafic de faible vitesse dans un flux de trafic de plus haute vitesse

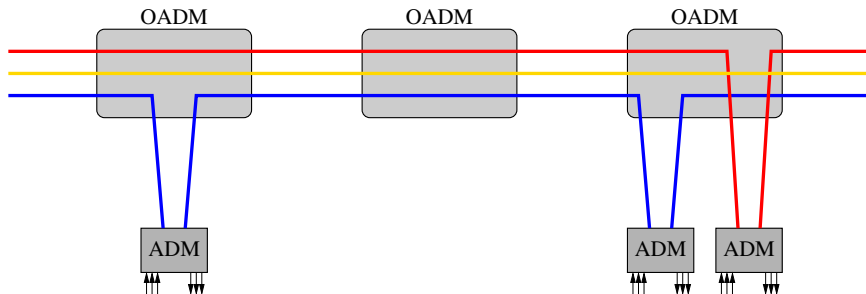
—→ on affecte plusieurs requêtes de faible vitesse à la même longueur d'onde (TDM, Time Division Multiplexing)

- Objectifs

- ▶ Partage efficace de la bande passante
- ▶ Réduire le coût des équipements de routage

ADM et OADM

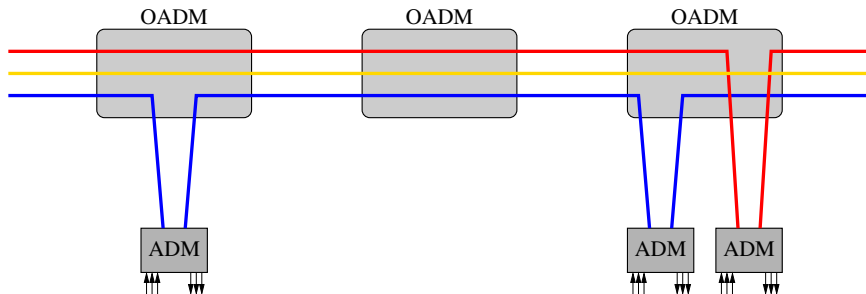
- **OADM** (Optical Add/Drop Multiplexer)= insérer et extraire une longueur d'onde d'une fibre optique
- **ADM** (Add/Drop Multiplexer)= insérer et extraire OC/STM (signal électrique de faible vitesse) d'une longueur d'onde



→ on veut minimiser le nombre d'ADMs

ADM et OADM

- **OADM** (Optical Add/Drop Multiplexer)= insérer et extraire une longueur d'onde d'une fibre optique
- **ADM** (Add/Drop Multiplexer)= insérer et extraire OC/STM (signal électrique de faible vitesse) d'une longueur d'onde



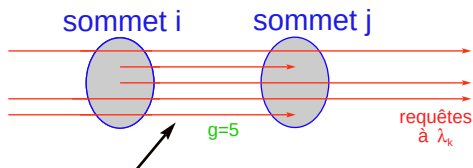
→ on veut minimiser le nombre d'ADM

Définitions

- **Requête** (i, j) : une paire de sommets (i, j) qui veulent faire un échange de trafic (de faible vitesse)

Définitions

- **Requête** (i, j) : une paire de sommets (i, j) qui veulent faire un échange de trafic (de faible vitesse)
- **Facteur de groupage** g :



Pour chaque longueur d'onde et chaque arc entre 2 sommets, il peut y avoir au plus **g** requêtes qui utilisent cet arc sur cette longueur d'onde

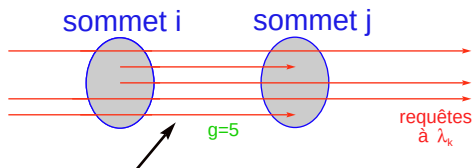
Exemple:

Capacité d'une long. d'onde = 2400 Mb/s

Vitesse d'une requête = 600 Mb/s $\Rightarrow g = 4$

Définitions

- **Requête** (i, j) : une paire de sommets (i, j) qui veulent faire un échange de trafic (de faible vitesse)
- **Facteur de groupage** g :



Pour chaque longueur d'onde et chaque arc entre 2 sommets, il peut y avoir au plus **g** requêtes qui utilisent cet arc sur cette longueur d'onde

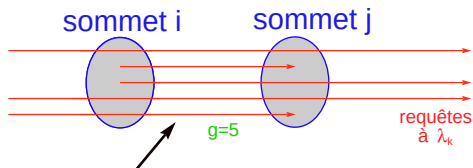
Exemple:

Capacité d'une long. d'onde = 2400 Mb/s

Vitesse d'une requête = 600 Mb/s $\Rightarrow g = 4$

Definitions

- **Requête** (i, j) : une paire de sommets (i, j) qui veulent faire un échange de trafic (de faible vitesse)
- **Facteur de groupage** g :

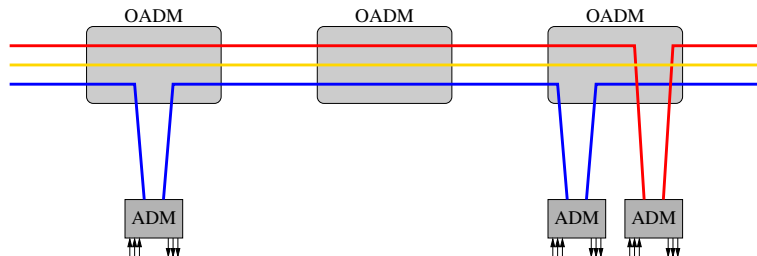


Pour chaque longueur d'onde et chaque arc entre 2 sommets, il peut y avoir au plus **g** requêtes qui utilisent cet arc sur cette longueur d'onde

- **Charge (load)** d'un arc sur une long. d'onde: le nombre de requêtes qui utilisent cet arc sur cette longueur d'onde ($\leq g$)

ADM et OADM

- **OADM** (Optical Add/Drop Multiplexer)= insérer et extraire une longueur d'onde d'une fibre optique
- **ADM** (Add/Drop Multiplexer)= insérer et extraire OC/STM (signal électrique de faible vitesse) d'une longueur d'onde



- **Idée:** Utiliser **ADM** seulement au sommet initial et au sommet final d'une *requête* (lightpaths) pour économiser le maximum nombre d'ADM

Pour fixé les idées...

- Modèle:

Topologie	→	graphe orienté G
Ensemble des requêtes	→	graphe orienté R
Facteur de groupage	→	entier g
Requêtes sur 1 long. d'onde	→	arêtes d'un sous-graphe de R
ADM sur 1 long. d'onde	→	sommet d'un sous-graphe de R

- On va étudier le cas où $G = C_n$ (anneau)

Pour fixé les idées...

- Modèle:

Topologie	→	graphe orienté G
Ensemble des requêtes	→	graphe orienté R
Facteur de groupage	→	entier g
Requêtes sur 1 long. d'onde	→	arêtes d'un sous-graphe de R
ADM sur 1 long. d'onde	→	sommet d'un sous-graphe de R

- On va étudier le cas où $G = C_n$ (anneau)

Définition du problème

Groupage de trafic sur l'anneau

- Entrées**
- Un cycle C_n à n sommets (réseau)
 - Un graphe orienté R (ensemble de requêtes)
 - Un facteur de groupage g
- Sorties**
- Trouver pour chaque arc $r \in R$ un chemin $P(r)$ dans C_n , et une partition des arêtes de R en sous-graphes R_ω , $1 \leq \omega \leq W$, de telle sorte que

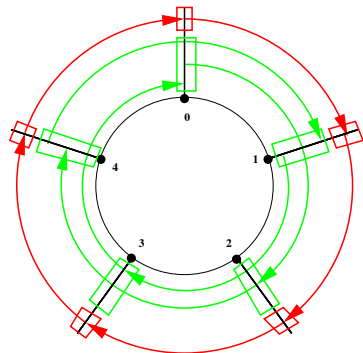
$$\forall e \in E(C_n), \quad \text{load}(R_\omega, e) \leq g$$

- Objectif**
- Minimiser $\sum_{\omega=1}^W |V(R_\omega)|$

Exemple: $n = 5$, $R = K_5$, et $g = 2$

Exemple: $n = 5$, $R = K_5$, et $g = 2$

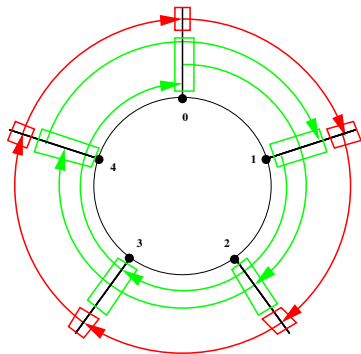
On décompose ici R de deux manières, en utilisant deux long. d'onde:



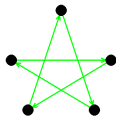
10 ADMS

Exemple: $n = 5$, $R = K_5$, et $g = 2$

On décompose ici R de deux manières, en utilisant deux long. d'onde:

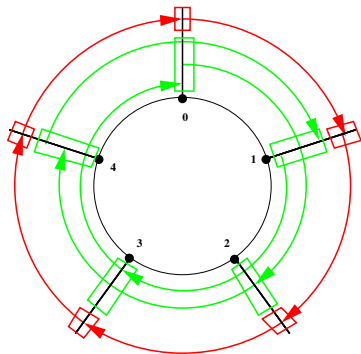


10 ADMS

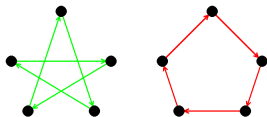


Exemple: $n = 5$, $R = K_5$, et $g = 2$

On décompose ici R de deux manières, en utilisant deux long. d'onde:

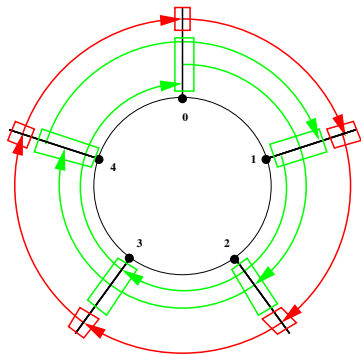


10 ADMS

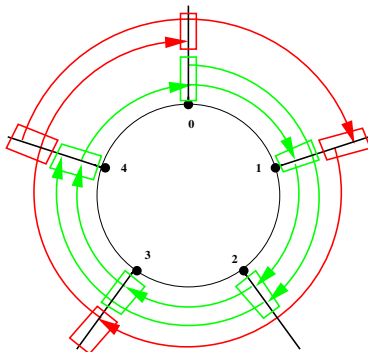


Exemple: $n = 5$, $R = K_5$, et $g = 2$

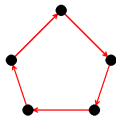
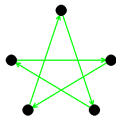
On décompose ici R de deux manières, en utilisant deux long. d'onde:



10 ADMS

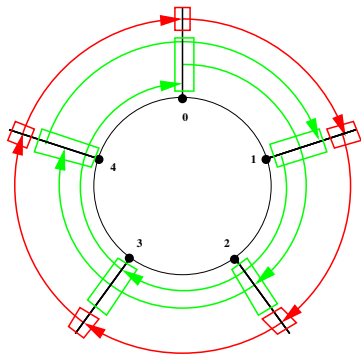


9 ADMS

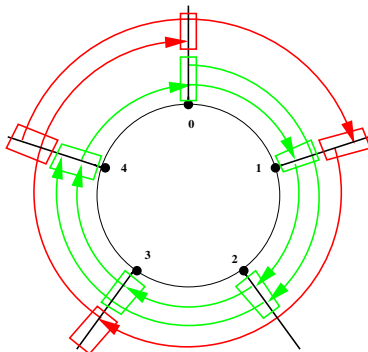


Exemple: $n = 5$, $R = K_5$, et $g = 2$

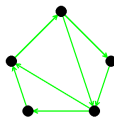
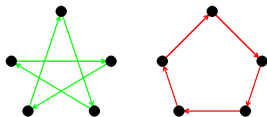
On décompose ici R de deux manières, en utilisant deux long. d'onde:



10 ADMS

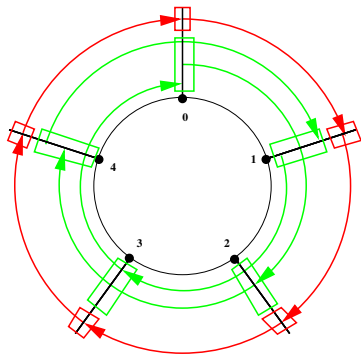


9 ADMS

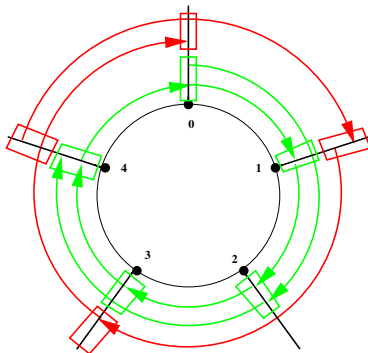
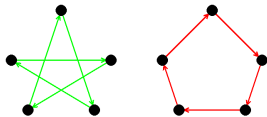


Exemple: $n = 5$, $R = K_5$, et $g = 2$

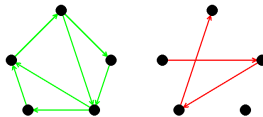
On décompose ici R de deux manières, en utilisant deux long. d'onde:



10 ADMS



9 ADMS



État de l'art (sur l'anneau)

- Complexité:

- ▶ TRAFFIC GROOMING (TG) est **NP-complet** quand g fait partie de l'entrée
(*T.Chow et P.Lin, Networks'04*)
- ▶ TG reste NP-complet aussi pour $g \geq 1$ fixé
(*M.Shalom, W.Unger et S.Zaks, FUN'07*)
- ▶ Si g fait partie de l'entrée, TG **n'est pas dans APX**
(*S.Huang, R.Dutta et G.N.Rouskas, IEEE JSAC'06*)
- ▶ **Problème ouvert:** inapproximabilité quand g est fixé

- Approximabilité:

- ▶ Trouver une $\sqrt{g}$ -approximation est trivial (en temps polynomial en n et en g)
- ▶ Le meilleur algo d'approx. a ratio $\mathcal{O}(\log g)$, mais le temps d'exécution est exponentiel en g (n^g)
(*M.Flammini et al., ISAAC'05*)
- ▶ **Problème ouvert:** bon algo en temps polynomial en g et n (c.a.d., algo quand g fait aussi partie de l'entrée)

État de l'art (sur l'anneau)

- Complexité:

- ▶ TRAFFIC GROOMING (TG) est **NP-complet** quand g fait partie de l'entrée
(*T.Chow et P.Lin, Networks'04*)
- ▶ TG reste NP-complet aussi pour $g \geq 1$ fixé
(*M.Shalom, W.Unger et S.Zaks, FUN'07*)
- ▶ Si g fait partie de l'entrée, TG **n'est pas dans APX**
(*S.Huang, R.Dutta et G.N.Rouskas, IEEE JSAC'06*)
- ▶ **Problème ouvert:** inapproximabilité quand g est fixé

- Approximabilité:

- ▶ Trouver une $\sqrt{g}$ -approximation est trivial (en temps polynomial en n et en g)
- ▶ Le meilleur algo d'approx. a ratio $\mathcal{O}(\log g)$, mais le temps d'exécution est exponentiel en g (n^g)
(*M.Flammini et al., ISAAC'05*)
- ▶ **Problème ouvert:** bon algo en temps polynomial en g et n (c.a.d., algo quand g fait aussi partie de l'entrée)

État de l'art (sur l'anneau)

- Complexité:

- ▶ TRAFFIC GROOMING (TG) est **NP-complet** quand g fait partie de l'entrée
(*T.Chow et P.Lin, Networks'04*)
- ▶ TG reste NP-complet aussi pour $g \geq 1$ fixé
(*M.Shalom, W.Unger et S.Zaks, FUN'07*)
- ▶ Si g fait partie de l'entrée, TG **n'est pas dans APX**
(*S.Huang, R.Dutta et G.N.Rouskas, IEEE JSAC'06*)

- ▶ **Problème ouvert:** inapproximabilité quand g est fixé

- Approximabilité:

- ▶ Trouver une $\sqrt{g}$ -approximation est trivial (en temps polynomial en n et en g)
- ▶ Le meilleur algo d'approx. a ratio $\mathcal{O}(\log g)$, mais le temps d'exécution est exponentiel en g (n^g)
(*M.Flammini et al., ISAAC'05*)
- ▶ **Problème ouvert:** bon algo en temps polynomial en g et n (c.a.d., algo quand g fait aussi partie de l'entrée)

État de l'art (sur l'anneau)

- Complexité:

- ▶ TRAFFIC GROOMING (TG) est **NP-complet** quand g fait partie de l'entrée
(*T.Chow et P.Lin, Networks'04*)
- ▶ TG reste NP-complet aussi pour $g \geq 1$ fixé
(*M.Shalom, W.Unger et S.Zaks, FUN'07*)
- ▶ Si g fait partie de l'entrée, TG **n'est pas dans APX**
(*S.Huang, R.Dutta et G.N.Rouskas, IEEE JSAC'06*)
- ▶ **Problème ouvert:** inapproximabilité quand g est fixé

- Approximabilité:

- ▶ Trouver une $\sqrt{g}$ -approximation est trivial (en temps polynomial en n et en g)
- ▶ Le meilleur algo d'approx. a ratio $\mathcal{O}(\log g)$, mais le temps d'exécution est exponentiel en g (n^g)
(*M.Flammini et al., ISAAC'05*)
- ▶ **Problème ouvert:** bon algo en temps polynomial en g et n (c.a.d., algo quand g fait aussi partie de l'entrée)

État de l'art (sur l'anneau)

- Complexité:

- ▶ TRAFFIC GROOMING (TG) est **NP-complet** quand g fait partie de l'entrée
(*T.Chow et P.Lin, Networks'04*)
- ▶ TG reste NP-complet aussi pour $g \geq 1$ fixé
(*M.Shalom, W.Unger et S.Zaks, FUN'07*)
- ▶ Si g fait partie de l'entrée, TG **n'est pas dans APX**
(*S.Huang, R.Dutta et G.N.Rouskas, IEEE JSAC'06*)
- ▶ **Problème ouvert:** inapproximabilité quand g est fixé

- Approximabilité:

- ▶ Trouver une $\sqrt{g}$ -approximation est trivial (en temps polynomial en n et en g)
- ▶ Le meilleur algo d'approx. a ratio $\mathcal{O}(\log g)$, mais le temps d'exécution est exponentiel en g (n^g)
(*M.Flammini et al., ISAAC'05*)
- ▶ **Problème ouvert:** bon algo en temps polynomial en g et n (c.a.d., algo quand g fait aussi partie de l'entrée)

État de l'art (sur l'anneau)

- Complexité:

- ▶ TRAFFIC GROOMING (TG) est **NP-complet** quand g fait partie de l'entrée
(*T.Chow et P.Lin, Networks'04*)
- ▶ TG reste NP-complet aussi pour $g \geq 1$ fixé
(*M.Shalom, W.Unger et S.Zaks, FUN'07*)
- ▶ Si g fait partie de l'entrée, TG **n'est pas dans APX**
(*S.Huang, R.Dutta et G.N.Rouskas, IEEE JSAC'06*)
- ▶ **Problème ouvert:** inapproximabilité quand g est fixé

- Approximabilité:

- ▶ Trouver une $\sqrt{g}$ -approximation est trivial (en temps polynomial en n et en g)
- ▶ Le meilleur algo d'approx. a ratio $\mathcal{O}(\log g)$, mais le temps d'exécution est exponentiel en g (n^g)
(*M.Flammini et al., ISAAC'05*)
- ▶ **Problème ouvert:** *bon* algo en temps polynomial en g et n (c.a.d., algo quand g fait aussi partie de l'entrée)

État de l'art (sur l'anneau)

- Complexité:

- ▶ TRAFFIC GROOMING (TG) est **NP-complet** quand g fait partie de l'entrée
(*T.Chow et P.Lin, Networks'04*)
- ▶ TG reste NP-complet aussi pour $g \geq 1$ fixé
(*M.Shalom, W.Unger et S.Zaks, FUN'07*)
- ▶ Si g fait partie de l'entrée, TG **n'est pas dans APX**
(*S.Huang, R.Dutta et G.N.Rouskas, IEEE JSAC'06*)
- ▶ **Problème ouvert:** inapproximabilité quand g est fixé

- Approximabilité:

- ▶ Trouver une $\sqrt{g}$ -approximation est trivial (en temps polynomial en n et en g)
- ▶ Le meilleur algo d'approx. a ratio $\mathcal{O}(\log g)$, mais le temps d'exécution est exponentiel en g (n^g)
(*M.Flammini et al., ISAAC'05*)
- ▶ **Problème ouvert:** bon algo en temps polynomial en g et n (c.a.d., algo quand g fait aussi partie de l'entrée)

Inapproximabilité du groupage pour g fixé

- Ici on va présenter la réponse à la première question ouverte:

Theorem

Le groupage de trafic sur l'anneau est APX-complet pour $g \geq 1$ fixé

Theorem

Le groupage de trafic sur le chemin est APX-complet pour $g \geq 2$ fixé

- Pour prouver ces résultats, on réduit le groupage au problème suivant:

trouver le nombre maximum de triangles arête-disjoints dans un graphe

Inapproximabilité du groupage pour g fixé

- Ici on va présenter la réponse à la première question ouverte:

Theorem

Le groupage de trafic sur l'anneau est APX-complet pour $g \geq 1$ fixé

Theorem

Le groupage de trafic sur le chemin est APX-complet pour $g \geq 2$ fixé

- Pour prouver ces résultats, on réduit le groupage au problème suivant:

trouver le nombre maximum de triangles arête-disjoints dans un graphe

Inapproximabilité du groupage pour g fixé

- Ici on va présenter la réponse à la première question ouverte:

Theorem

Le groupage de trafic sur l'anneau est APX-complet pour $g \geq 1$ fixé

Theorem

Le groupage de trafic sur le chemin est APX-complet pour $g \geq 2$ fixé

- Pour prouver ces résultats, on réduit le groupage au problème suivant:

trouver le nombre maximum de triangles arête-disjoints dans un graphe

MECT-B

- MAXIMUM BOUNDED EDGE COVERING BY TRIANGLES (MECT-B):
trouver le nombre maximum de triangles **arête-disjoints** dans un graphe de degré borné par B
- Le problème est NP-complet
(*I.Holyer, SIAM J.Comput'81*)
- Trouver triangles **sommet-disjoints** est APX-complet
(*V.Kann, Inf. Proces. Let'91*)
- On a prouvé que MECT-B est aussi APX-complet

MECT-B

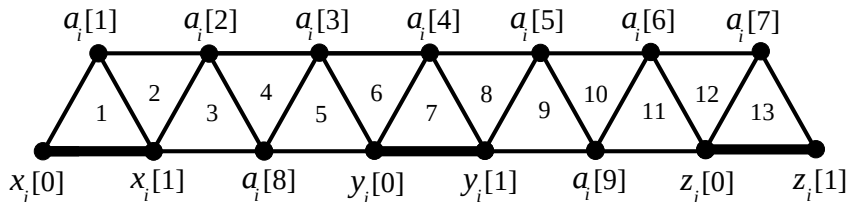
- MAXIMUM BOUNDED EDGE COVERING BY TRIANGLES (MECT-B):
trouver le nombre maximum de triangles **arête-disjoints** dans un graphe de degré borné par B
- Le problème est NP-complet
(*I.Holyer, SIAM J.Comput'81*)
- Trouver triangles **sommet-disjoints** est APX-complet
(*V.Kann, Inf. Proces. Let'91*)
- On a prouvé que MECT-B est aussi APX-complet

- MAXIMUM BOUNDED EDGE COVERING BY TRIANGLES (MECT-B):
trouver le nombre maximum de triangles **arête-disjoints** dans un graphe de degré borné par B
- Le problème est NP-complet
(*I.Holyer, SIAM J.Comput'81*)
- Trouver triangles **sommet-disjoints** est APX-complet
(*V.Kann, Inf. Proces. Let'91*)
- On a prouvé que MECT-B est aussi APX-complet

Idée de la preuve

Réduction de MAXIMUM BOUNDED COVERING BY 3-SETS (MAX 3SC-B):

- Pour chaque sous-ensemble $c_i = \{x_i, y_i, z_i\}$ on construit le graphe suivant:



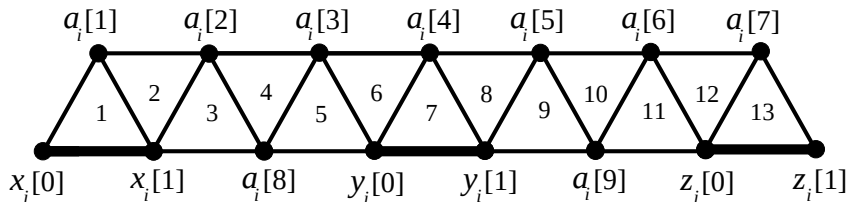
- On peut en déduire que

$$OPT(MECT-B) \leq (18B + 1)OPT(MAX\ 3SC-B)$$

Idée de la preuve

Réduction de MAXIMUM BOUNDED COVERING BY 3-SETS (MAX 3SC-B):

- Pour chaque sous-ensemble $c_i = \{x_i, y_i, z_i\}$ on construit le graphe suivant:



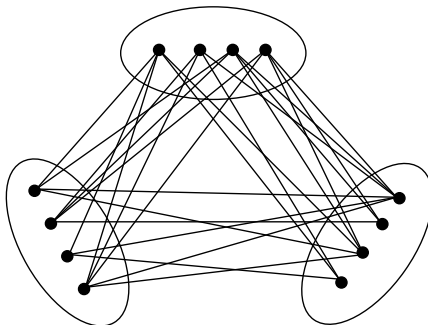
- On peut en déduire que

$$OPT(MECT-B) \leq (18B + 1)OPT(\text{MAX 3SC-B})$$

APX groupage anneau

Idée de la preuve pour $g = 1$:

- On prend un graphe de requêtes triparti

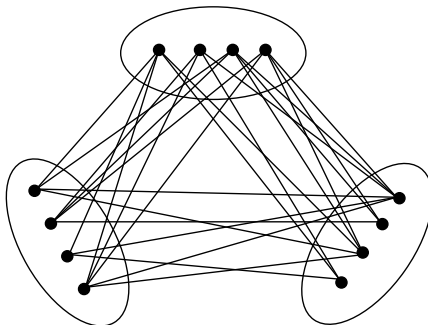


- Le problème de groupage de trafic revient à trouver le nombre maximum de triangles arête-disjoints
- Donc, le groupage est aussi APX-complet

APX groupage anneau

Idée de la preuve pour $g = 1$:

- On prend un graphe de requêtes triparti



- Le problème de groupage de trafic revient à trouver le nombre maximum de triangles arête-disjoints
- Donc, le groupage est aussi APX-complet

APX groupage sur le chemin

- Le problème est dans P pour $g = 1$
(*J-C.Bermond, L.Braud et D.Coudert, SIROCCO'05*)
- La complexité pour $g \geq 2$ fixé (même si P/NP?) était un problème ouvert
- On a prouvé que le problème est APX-complet pour $g \geq 2$ fixé
- Qu'est-ce qu'on peut dire sur l'approximabilité (anneau/chemin)...?

APX groupage sur le chemin

- Le problème est dans P pour $g = 1$
(*J-C.Bermond, L.Braud et D.Coudert, SIROCCO'05*)
- La complexité pour $g \geq 2$ fixé (même si P/NP?) était un problème ouvert
- On a prouvé que le problème est APX-complet pour $g \geq 2$ fixé
- Qu'est-ce qu'on peut dire sur l'approximabilité (anneau/chemin)...?

APX groupage sur le chemin

- Le problème est dans P pour $g = 1$
(*J-C.Bermond, L.Braud et D.Coudert, SIROCCO'05*)
- La complexité pour $g \geq 2$ fixé (même si P/NP?) était un problème ouvert
- On a prouvé que le problème est APX-complet pour $g \geq 2$ fixé
- Qu'est-ce qu'on peut dire sur l'approximabilité (anneau/chemin)...?

Trouver un sous-graphe dense

Définition du problème

- MINIMUM SUBGRAPH OF MINIMUM DEGREE $\geq d$ (MSMD_{*d*}):
Étant donné un graphe, trouver le plus petit sous-graphe induit de degré minimum au moins *d*
- Le problème pour $d = 2$ est trouver la maille d'un graphe (longueur du plus court cycle), qui est polynomial
- On a prouvé que pour $d \geq 3$, MSMD_{*d*} n'est pas dans APX

Définition du problème

- MINIMUM SUBGRAPH OF MINIMUM DEGREE $\geq d$ (MSMD_{*d*}):
Étant donné un graphe, trouver le plus petit sous-graphe induit de degré minimum au moins *d*
- Le problème pour $d = 2$ est trouver la maille d'un graphe (longueur du plus court cycle), qui est polynomial
- On a prouvé que pour $d \geq 3$, MSMD_{*d*} n'est pas dans APX

Définition du problème

- MINIMUM SUBGRAPH OF MINIMUM DEGREE $\geq d$ (MSMD $_d$):
Étant donné un graphe, trouver le plus petit sous-graphe induit de degré minimum au moins d
- Le problème pour $d = 2$ est trouver la maille d'un graphe (longueur du plus court cycle), qui est polynomial
- On a prouvé que pour $d \geq 3$, MSMD $_d$ n'est pas dans APX

Relation avec le groupage

- En gros, le groupage consiste à trouver des sous-graphes avec un grand rapport $\frac{\text{arêtes}}{\text{sommets}}$ (**densité**), et un nombre "borné" d'arêtes
- C.a.d., on veut trouver sous-graphes avec un grand degré moyen (et nombre "borné" d'arêtes)
- La densité et le degré minimum sont reliés par un facteur 2
- Et donc, si on sait trouver de petits sous-graphes de degré minimum fixé, on sait aussi trouver de petits sous-graphes ayant une "bonne" densité

Relation avec le groupage

- En gros, le groupage consiste à trouver des sous-graphes avec un grand rapport $\frac{\text{arêtes}}{\text{sommets}}$ (**densité**), et un nombre "borné" d'arêtes
- C.a.d., on veut trouver sous-graphes avec un grand degré moyen (et nombre "borné" d'arêtes)
- La densité et le degré minimum sont reliés par un facteur 2
- Et donc, si on sait trouver de petits sous-graphes de degré minimum fixé, on sait aussi trouver de petits sous-graphes ayant une "bonne" densité

Idée de la preuve pour $d = 3$

- **Gap-preserving reduction** de VERTEX COVER:

Instance H de VERTEX COVER $\rightarrow$ Instance G de MSMD₃

- On construit un arbre ternaire T avec une feuille pour chaque arête de H
- On rajoute les sommets de H et les relations d'adjacence avec les arêtes (plus un cycle...)
- MSMD₃ dans G revient à VERTEX COVER dans H

Idée de la preuve pour $d = 3$

- **Gap-preserving reduction** de VERTEX COVER:

Instance H de VERTEX COVER $\rightarrow$ Instance G de MSMD₃

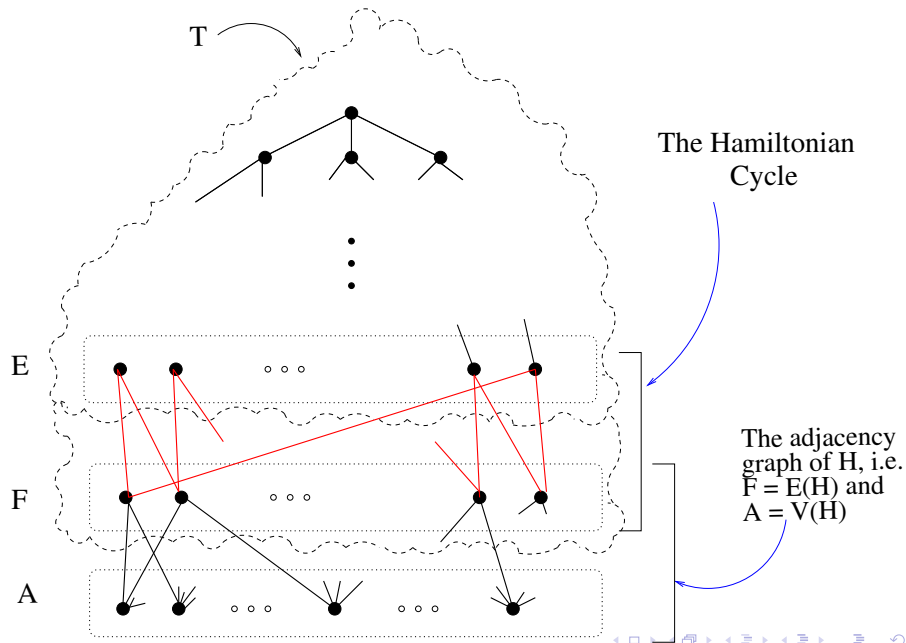
- On construit un arbre ternaire T avec une feuille pour chaque arête de H
- On rajoute les sommets de H et les relations d'adjacence avec les arêtes (plus un cycle...)
- MSMD₃ dans G revient à VERTEX COVER dans H

Idée de la preuve pour $d = 3$

- **Gap-preserving reduction** de VERTEX COVER:

Instance H de VERTEX COVER $\rightarrow$ Instance G de MSMD₃

- On construit un arbre ternaire T avec une feuille pour chaque arête de H
- On rajoute les sommets de H et les relations d'adjacence avec les arêtes (plus un cycle...)
- MSMD₃ dans G revient à VERTEX COVER dans H



Idée de la preuve (3)

- Ça prouve qu'il n'y a pas de PTAS
- On fait une amplification d'erreur pour prouver qu'il n'y a pas d'algorithme à facteur constant...

Conclusions

● Groupage de trafic:

- ▶ On a prouvé que le groupage sur l'anneau est APX-complet pour $g \geq 1$ fixé
- ▶ On a prouvé que le groupage sur le chemin est APX-complet pour $g \geq 2$ fixé
- ▶ On a des algorithmes d'approximation pour le groupage sur l'anneau en temps polynomial en n et en g , avec un rapport

$$\mathcal{O}(n^{1/3} \log^2 n) \quad \text{pour } g \geq 1 \quad \text{non fixé.}$$

- ▶ On conjecture qu'il existe $\varepsilon > 0$ tel que le groupage de trafic est difficile à approximer à un facteur n^ε

Conclusions

● Groupage de trafic:

- ▶ On a prouvé que le groupage sur l'anneau est APX-complet pour $g \geq 1$ fixé
- ▶ On a prouvé que le groupage sur le chemin est APX-complet pour $g \geq 2$ fixé
- ▶ On a des algorithmes d'approximation pour le groupage sur l'anneau en temps polynomial en n et en g , avec un rapport

$$\mathcal{O}(n^{1/3} \log^2 n) \quad \text{pour } g \geq 1 \quad \text{non fixé.}$$

- ▶ On conjecture qu'il existe $\varepsilon > 0$ tel que le groupage de trafic est difficile à approximer à un facteur n^ε

Conclusions

● Groupage de trafic:

- ▶ On a prouvé que le groupage sur l'anneau est APX-complet pour $g \geq 1$ fixé
- ▶ On a prouvé que le groupage sur le chemin est APX-complet pour $g \geq 2$ fixé
- ▶ On a des algorithmes d'approximation pour le groupage sur l'anneau en temps polynomial en n et en g , avec un rapport

$$\mathcal{O}(n^{1/3} \log^2 n) \quad \text{pour } g \geq 1 \quad \text{non fixé.}$$

- ▶ On conjecture qu'il existe $\varepsilon > 0$ tel que le groupage de trafic est difficile à approximer à un facteur n^ε

Conclusions (2)

- **Trouver le plus petit sous-graphe de degré minimum au moins d :**
 - ▶ On a prouvé que MSMD_d , $d \geq 3$, n'est pas dans APX
 - ▶ On sait que MSMD_d , $d \geq 3$, est $W[1]$ -dur, et donc on ne peut pas espérer trouver des algorithmes FPT pour un graphe quelconque
 - ▶ On a des algorithmes FPT pour la classe des graphes mineur-exclus
(par exemple: graphes planaires, graphes à treewidth localement bornée, graphes de genre borné...)

Conclusions (2)

- **Trouver le plus petit sous-graphe de degré minimum au moins d :**
 - ▶ On a prouvé que MSMD_d , $d \geq 3$, n'est pas dans APX
 - ▶ On sait que MSMD_d , $d \geq 3$, est $W[1]$ -dur, et donc on ne peut pas espérer trouver des algorithmes FPT pour un graphe quelconque
 - ▶ On a des algorithmes FPT pour la classe des graphes mineur-exclus
(par exemple: graphes planaires, graphes à treewidth localement bornée, graphes de genre borné...)

Conclusions (2)

- **Trouver le plus petit sous-graphe de degré minimum au moins d :**
 - ▶ On a prouvé que MSMD_d , $d \geq 3$, n'est pas dans APX
 - ▶ On sait que MSMD_d , $d \geq 3$, est $W[1]$ -dur, et donc on ne peut pas espérer trouver des algorithmes FPT pour un graphe quelconque
 - ▶ On a des algorithmes FPT pour la classe des graphes mineur-exclus
(par exemple: graphes planaires, graphes à treewidth localement bornée, graphes de genre borné...)

Merci!