

## **Routage efficace en énergie**

Frédéric Giroire, Dorian Mazauric, Joanna Moulierac

► **To cite this version:**

Frédéric Giroire, Dorian Mazauric, Joanna Moulierac. Routage efficace en énergie. Ducourthial, Bertrand et Felber, Pascal. 13es Rencontres Francophones sur les Aspects Algorithmiques de Télécommunications (AlgoTel), 2011, Cap Estérel, France. 2011. <inria-00587944>

**HAL Id: inria-00587944**

**<https://hal.inria.fr/inria-00587944>**

Submitted on 21 Apr 2011

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

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

# Routage efficace en énergie<sup>†</sup>

Frédéric Giroire and Dorian Mazauric and Joanna Moulrierac

MASCOTTE, I3S(CNRS/UNS), INRIA, Sophia Antipolis, France

---

De récentes études montrent que la charge de trafic des routeurs n'a qu'une faible influence sur leur consommation énergétique. Par conséquent, la consommation dans les réseaux est fortement liée au nombre d'équipements du réseau activés (interfaces, chassis, etc). Dans un objectif de minimisation de l'énergie dans les réseaux, il est intéressant de minimiser le nombre (pondéré) d'équipements utilisés lors du routage. Dans cet article, nous considérons une architecture simplifiée où un lien entre deux routeurs relie deux interfaces. Quand un lien n'est pas activé, les deux interfaces correspondantes peuvent être éteintes. Par conséquent, afin de réduire la consommation d'énergie, l'objectif est de trouver un routage qui minimise le nombre de liens utilisés et satisfait toutes les demandes. Nous montrons des résultats d'inapproximabilité de ce problème, même si l'on considère des instances particulières. Nous prouvons des bornes en général et pour des topologies particulières telles que la grille, l'arbre ou le graphe complet. Nous proposons ensuite une heuristique dont nous évaluons les performances à l'aide de simulations sur des topologies réelles. Nous étudions ensuite l'impact de ces solutions efficaces en énergie sur la tolérance aux pannes et sur la longueur moyenne des routes.

**Keywords:** Minimisation d'énergie, routage, graphes, programmation linéaire, algorithmes

---

## 1 Introduction

Les technologies d'information et de communication (ICT) sont responsables à elles seules de 2% à 10% (selon les estimations) de la consommation mondiale [CLM09]. Dans ce papier, nous nous intéressons à la consommation liée aux réseaux et plus particulièrement à celle générée par le routage des demandes. Il est estimé que les hubs, switches et routeurs consomment 6 TWh par an aux USA [NC05]. Des études récentes mettent en exergue le fait que la charge de trafic d'un routeur a peu d'influence sur sa consommation d'énergie [CSB<sup>+</sup>08]. Ainsi, la consommation dépend principalement du nombre d'équipements du réseau allumés. Il s'agit alors de trouver un routage des demandes qui minimise le nombre d'équipements utilisés.

Néanmoins dans la plupart des réseaux, les routeurs ne peuvent pas être éteints. En effet, ils sont parfois source et/ou destination des demandes ; ils sont également parfois indispensables en cas de panne dans le réseau. Pour ces raisons, nous considérons une architecture simplifiée dans laquelle une connexion entre deux routeurs est représentée par un lien reliant deux interfaces. Il est possible alors d'économiser de l'énergie en éteignant les deux interfaces qui sont aux deux extrémités d'un lien. Ainsi, nous pouvons modéliser le réseau par un graphe et l'objectif est alors de *trouver un sous-graphe minimum en nombre d'arêtes permettant de router les demandes*.

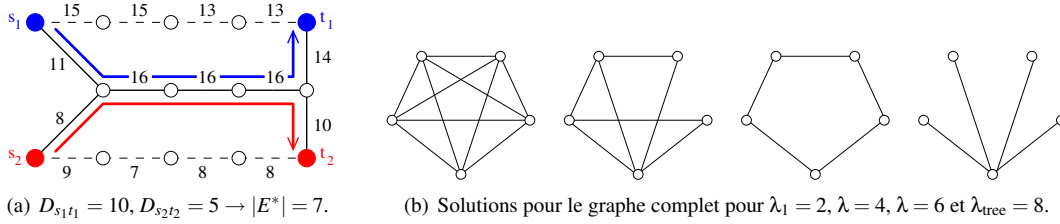
**Travaux existants.** Dans [CSB<sup>+</sup>08], les auteurs proposent un programme linéaire pour ce problème. Ils analysent expérimentalement et par simulations la quantité d'énergie qui peut être sauvée pour certains réseaux. Dans [PVC<sup>+</sup>09], les auteurs étudient l'impact de la technologie sur l'efficacité en énergie du routage. Dans [GS03], différentes techniques sont proposées pour économiser de l'énergie comme par exemple adapter le débit en fonction du trafic. Dans [CNR10], les auteurs proposent une modulation de la configuration radio dans les réseaux sans-fil à large bande pour réduire la consommation.

## 2 Modélisation

Le réseau est modélisé par un graphe pondéré  $G = (V, E)$  avec  $c_e$  représentant la capacité de l'arête  $e \in E$ . L'ensemble des demandes est  $D = \{\mathcal{D}_{st} > 0; (s, t) \in V \times V, s \neq t\}$  avec  $\mathcal{D}_{st}$  le volume de trafic de  $s$  à  $t$ . Une demande  $\mathcal{D}_{st}$  doit être routée via un chemin élémentaire de  $s$  à  $t$ . Un routage valide des demandes est un ensemble de chemins dans  $G$  pour chacune des demandes  $\mathcal{D}_{st} \in D$  tel que pour chaque arête  $e \in E$ , le

---

<sup>†</sup>ANR DIMAGREEN. La version complète de notre travail se trouve dans le rapport de recherche [GMMO10]



**FIG. 1:** (a) Solution optimale pour une instance du MINIMUM EDGES ROUTING PROBLEM et (b) évolution de la solution optimale pour le graphe complet à 5 sommets avec un trafic *all-to-all* en fonction de  $\lambda$ .

volume total de trafic passant par  $e$  n'excède pas la capacité  $c_e$ . Un problème de décision classique est de déterminer si un tel routage existe ou non. Formellement :

**Définition 1.** Etant donné un graphe pondéré  $G = (V, E)$  et un ensemble de demandes  $D$ , le ROUTING PROBLEM consiste à décider si il existe un routage valide des demandes de  $D$  dans  $G$ .

Le problème que nous étudions a la contrainte supplémentaire de trouver un routage minimisant le nombre d'arêtes utilisées. Formellement :

**Définition 2.** Etant donné un graphe pondéré  $G = (V, E)$  et un ensemble de demandes  $D$ , le MINIMUM EDGES ROUTING PROBLEM consiste à trouver un ensemble de cardinalité minimum  $E^* \subseteq E$  tel que il existe un routage valide des demandes  $D$  dans  $G^* = (V, E^*)$ .

Le MINIMUM EDGES ROUTING PROBLEM peut être modélisé comme un problème de flots entier à plusieurs commodités.

**Figure 1(a).** Dans cet exemple, il y a deux demandes  $D_{s_1 t_1} = 10$  et  $D_{s_2 t_2} = 5$ . La solution  $E^*$  pour le MINIMUM EDGES ROUTING PROBLEM est composée de  $|E^*| = 7$  arêtes. Cet exemple très simple montre que le routage par plus courts chemins ne donnent pas (nécessairement) une solution optimale. De plus, le nombre d'arêtes de la solution donnée par les plus courts chemins peut être arbitrairement éloigné de  $E^*$ . Enfin, remarquons que si par exemple  $D_{s_1 t_1} = 12$ , alors le routage par les plus courts chemins est optimal.

## 3 Résultats

### 3.1 Inapproximabilité

Le problème de décision ROUTING PROBLEM est NP-complet même lorsqu'il y a seulement deux demandes [EIS75]. Le problème de minimisation MINIMUM EDGES ROUTING PROBLEM est NP-difficile [GP90]. Nous montrons que le MINIMUM EDGES ROUTING PROBLEM n'est pas dans APX. En effet si  $E^*$  est une solution optimale alors  $\forall k \geq 1$ , il n'existe pas d'algorithme polynomial garantissant  $|E'|/|E^*| \leq k$  où  $E'$  est la solution donnée par l'algorithme d'approximation (à moins que  $P=NP$ ). Les théorèmes 1 et 2 montrent cela même pour des instances très particulières.

**Théorème 1.** Le problème n'est pas dans APX même si il n'y a que deux demandes.

**Théorème 2.** Le problème n'est pas dans APX même si toutes les arêtes ont une capacité constante  $c$ .

### 3.2 Heuristique

Nous proposons dans cette section une heuristique pour le MINIMUM EDGES ROUTING PROBLEM. LESS LOADED EDGE HEURISTIC commence par trouver un routage valide dans  $G = (V, E)$  en utilisant une heuristique routant les demandes de manière gloutonne par des plus courts chemins. La métrique utilisée est  $c(e)/r(e)$  où  $r(e)$  désigne la capacité résiduelle. Initialement  $r(e) = c(e), \forall e \in E$ . L'idée est ensuite de supprimer l'arête  $e \in E$  dont le ratio  $c(e)/r(e)$  est le plus petit possible (dans un certain sens l'arête qui est la moins chargée). Nous recalculons un routage valide et procédons de la même façon pour diminuer le nombre d'arêtes. Si aucun routage valide n'est trouvé nous réinsérons l'arête supprimée et continuons le processus de suppression des arêtes. Notons qu'une arête ne peut pas être supprimée deux fois. Nous obtenons finalement un ensemble d'arêtes  $E'$  et un routage valide dans  $G' = (V, E')$ .

			Facteur de surdimensionnement OF				Arbre		Simulations		Valeurs obtenues avec les bornes			
	$ V $	$ E $	1	2	3	4	OF	Gain	$\lambda_1$	$\lambda_{tree}$	$\lambda_1$ Coupe	$\lambda_{tree}$ Borne	$\Delta$	$\lambda_{tree}$ Borne
Atlanta	15	22	0%	32%	36%	36%	2.66	36%	38	101	3	38	4	88
New York	16	49	2.0%	59%	63%	67%	5.2	69%	15	78	12	11	11	56
Nobel Germany	17	26	0%	35%	39%	39%	2.75	39%	44	121	4	36	5	104
France	25	45	0%	42%	44%	47%	3.13	47%	67	210	7	45	10	132
Norway	27	51	12%	43%	47%	47%	4.71	49%	75	354	6	61	6	220
Nobel EU	28	41	12%	32%	34%	34%	2.76	34%	131	362	3	131	5	264
Cost266	37	57	3.5%	32%	35%	37%	3.68	37%	175	644	4	171	4	540
Giul39	39	86	0%	45%	50%	52%	8.25	56%	85	702	11	70	8	340
Pioro40	40	89	0%	53%	54%	55%	5.12	56%	153	784	7	115	5	512
Zib54	54	80	0%	30%	33%	33%	4.71	34%	294	1385	6	243	10	576

(a) Interfaces économisées (en %).

(b) Évaluation de la charge donnée par les bornes.

**TAB. 1:** (a) Interfaces économisées (en %) selon la valeur du facteur de surdimensionnement  $OF$  et (b) Evaluation de  $\lambda_1$  et  $\lambda_{tree}$ . Ces deux valeurs sont comparées avec les bornes théoriques de la section 3.3.

### 3.3 Bornes théoriques

Nous prouvons des formules closes et des bornes pour ce ratio pour des topologies générales. Nous prouvons également des formules plus fines pour des topologies particulières (arbres, graphes complets, grilles). Nous présentons ici une petite partie de nos résultats.

Nous supposons dans cette section et dans le reste de l'article que, étant donné un graphe pondéré  $G = (V, E)$  représentant un réseau, chaque arête  $e \in E$  a une capacité constante  $c_e = c$ . De plus les demandes sont de type "all-to-all" avec  $\forall (s, t) \in V \times V, \mathcal{D}_{st} = \kappa$ . Nous posons  $\lambda = c/\kappa$ .

Nous cherchons notamment à calculer  $\lambda_1$  et  $\lambda_{tree}$ .  $\lambda_1$  est la plus petite valeur de  $\lambda$  telle que il existe un routage valide dans  $G$  (cela revient à trouver la plus petite capacité  $c$  correspondante).  $\lambda_{tree}$  est la plus petite valeur de  $\lambda$  telle que il existe un arbre dans  $G$  qui permette de router toutes les demandes.

**Figure 1(b).** Pour le graphe complet à 5 sommets,  $\lambda_1 = 2$  car chaque demande est routée via un plus court chemin (de taille 1 ici). En prenant un arbre formant une étoile, nous obtenons  $\lambda_{tree} = 8$ . Si nous avions pris un chemin par exemple, nous aurions obtenu  $\lambda = 12$ . Nous pouvons décrire la figure d'une autre manière : les 4 graphes représentent 4 solutions pour 4 valeurs de  $\lambda$ .

**Borne sur  $\lambda_1$ .** Étant donné  $S \subset V$ , nous notons  $\bar{S} = V \setminus S$ . Sous les hypothèses mentionnées précédemment et en utilisant un argument de coupe, nous donnons une borne sur  $\lambda_1 : c|E_{S\bar{S}}| \geq 2\kappa|S||\bar{S}|$  où  $|E_{S\bar{S}}|$  représente le nombre d'arêtes entre l'ensemble  $S$  et  $\bar{S}$ . En effet il y a une demande de volume  $\kappa$  entre tous les sommets de  $S$  et tous les sommets de  $\bar{S}$ , et ces demandes doivent nécessairement utiliser les arêtes de  $|E_{S\bar{S}}|$ . En trouvant le sous-ensemble  $S$  maximisant  $(2|S||\bar{S}|)/|E_{S\bar{S}}|$ , nous obtenons une borne inférieure sur  $\lambda_1$ . Nous verrons dans la section 3.4 que cette borne est très pertinente dans les réseaux réels.

**Borne sur  $\lambda_{tree}$ .** Soit  $\Delta$  le degré maximum dans  $G$ . Nous obtenons que  $\lambda_{tree} \geq 2 \left\lceil \frac{|V|-1}{\Delta} \right\rceil \left( |V| - \left\lceil \frac{|V|-1}{\Delta} \right\rceil \right)$ .

### 3.4 Simulations

Nous présentons dans cette section les résultats de simulations de notre heuristique. Pour cela nous utilisons dix réseaux classiques extraits de la librairie SNDLib (<http://sndlib.zib.de>). Nous regardons combien d'interfaces peuvent être éteintes selon la valeur de  $\lambda$ . Étant donnée une valeur  $\lambda$ , le facteur de surdimensionnement  $OF$  ("overprovisioning factor") correspondant est défini par le rapport  $\lambda/\lambda_1$ .

**Tableau 1 (a).** Nous donnons le pourcentage d'interfaces économisées pour différentes valeurs de  $OF$ . Nous remarquons que pour certaines topologies, des interfaces peuvent être économisées même pour  $OF = 1$  (12% pour *Norway* et *Nobel EU*). Lorsque  $\lambda$  double ( $OF = 2$ , représentant par exemple un trafic de nuit deux fois moins important), nous observons qu'environ un tiers des arêtes peuvent être économisées (et même 53% pour le réseau *Pioro40*). En augmentant encore  $OF$ , nous remarquons que le gain marginal devient faible. Les deux dernières colonnes mettent en exergue  $\lambda_{tree}$  et le pourcentage d'arêtes économisées.

**Tableau 1 (b).** Nous représentons la valeur  $\lambda_1$  correspondant à  $OF = 1$ . Comme décrit dans la section 3.3,  $\lambda_1$  dépend fortement de la coupe minimum du graphe. Nous calculons cette coupe minimum en utilisant un programme linéaire, puis nous en déduisons une borne inférieure pour  $\lambda_1$ . Nous remarquons que cette dernière est très proche (parfois identique) de la valeur obtenue par les simulations (même si cette dernière n'est pas toujours optimale). De façon analogue, nous obtenons par simulations la valeur  $\lambda_{tree}$ . Nous présentons

également le degré maximum  $\Delta$  de chaque graphe ainsi que la borne induite par celle-ci sur  $\lambda_{tree}$ . Nous remarquons ici des différences relativement importantes par rapport aux valeurs obtenues par simulations.

### 3.5 Impact des solutions proposées

**Impact sur la longueur des routes.** Nous remarquons que les routages obtenus ont un impact limité sur l'augmentation de la longueur des routes. Pour *Zib54*, l'accroissement est de 11% pour le cas extrême où le routage se fait avec un arbre (34% d'économie). En général, l'augmentation pour ce cas extrême varie de 11% à 50% avec une moyenne de 27%. Concernant le réseau le plus impacté *Giul39* avec un accroissement de 50% de la longueur des routes, l'économie est de 56%. Nous observons qu'une économie d'interfaces de 45% est atteinte pour  $OF = 2$  avec une augmentation de la longueur des routes de seulement 18%.

**Impact sur la tolérance aux pannes.** Nous mesurons l'impact sur la tolérance aux pannes en calculant le nombre moyen de chemins disjoints entre deux sommets. Ce nombre varie de 1.82 à 4.90 pour les réseaux complets. En routant les demandes via un arbre, ce nombre est 1 par définition (une seule route existe entre n'importe quelle paire de sommets). Cette valeur est presque atteinte avec  $OF = 3$ . Nous observons enfin que cette diminution est rapide car pour les dix réseaux, cette valeur est inférieure à 1.41 pour  $OF = 2$ .

## 4 Perspectives

D'un point de vue théorique, nous tentons actuellement de savoir si le problème de déterminer la valeur  $\lambda_{tree}$  est dans APX ou non. Autrement dit, nous pouvons facilement montrer (en utilisant des preuves analogues à celles des théorèmes 1 et 2) que déterminer  $\lambda_{tree}$  est NP-complet mais peut-on obtenir une approximation de cette valeur à une constante multiplicative près ?

D'un point de vue pratique, nous sommes en train d'expérimenter nos solutions sur une plateforme d'expérimentation composée d'une vingtaine de machines. Nous étudierons également ce problème en utilisant une architecture de réseau un peu plus complexe. Nous réfléchissons de plus à l'auto-adaptabilité de notre solution (déploiement, passage d'un routage à un autre, modification des protocoles de routage).

## Références

- [CLM09] L. Chiaraviglio, E. Leonardi, and M. Mellia. How much can Internet be greened? In *GreenComm*, 2009.
- [CNR10] D. Coudert, N. Nepomuceno, and H. Rivano. Power-efficient radio configuration in fixed broadband wireless networks. *Computer Communications*, 2010. To appear.
- [CSB<sup>+</sup>08] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsang, and S. Wright. Power awareness in network design and routing. In *IEEE INFOCOM*, pages 457–465, 2008.
- [EIS75] S. Even, A. Itai, and A. Shamir. On the complexity of time table and multi-commodity flow problems. In *16th Annual Symposium on Foundations of Computer Science, 1975.*, pages 184–193, 1975.
- [GMMO10] F. Giroire, D. Mazauric, J. Moulierac, and B. Onfroy. Minimizing routing energy consumption : from theoretical to practical results. Research Report 7234, INRIA, 05 2010.
- [GP90] G.M. Guisewite and P.M. Pardalos. Minimum concave-cost network flow problems : Applications, complexity, and algorithms. *Annals of Operations Research*, 25(1) :75–99, 1990.
- [GS03] M. Gupta and S. Singh. Greening of the internet. In *SIGCOMM*, 2003.
- [NC05] Bruce Nordman and Ken Christensen. Reducing the energy consumption of networked devices. [www.csee.usf.edu/~christen/energy/lbnl\\_talk.pdf](http://www.csee.usf.edu/~christen/energy/lbnl_talk.pdf), 2005.
- [PVC<sup>+</sup>09] B. Puype, W. Vereecken, D. Colle, M. Pickavet, and P. Demeester. Power Reduction Techniques in Multilayer Traffic Engineering. In *ICTON*, 2009.