Une introduction au codage de réseau aléatoire

Christine Bachoc

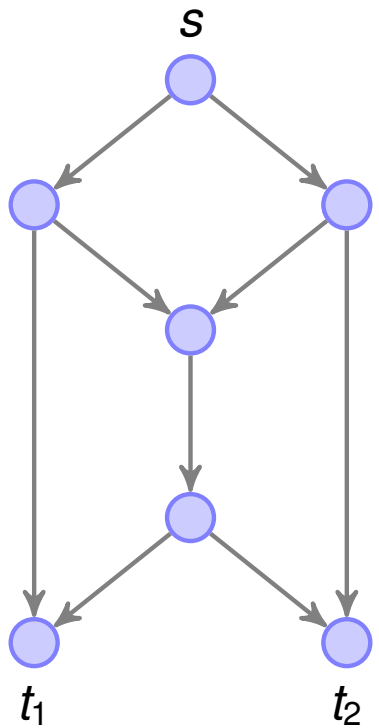
Université Bordeaux I, IMB

École de printemps Codage et Cryptographie
17 - 21 Mars 2014, Université Joseph Fourier, Grenoble

Réseau de communication

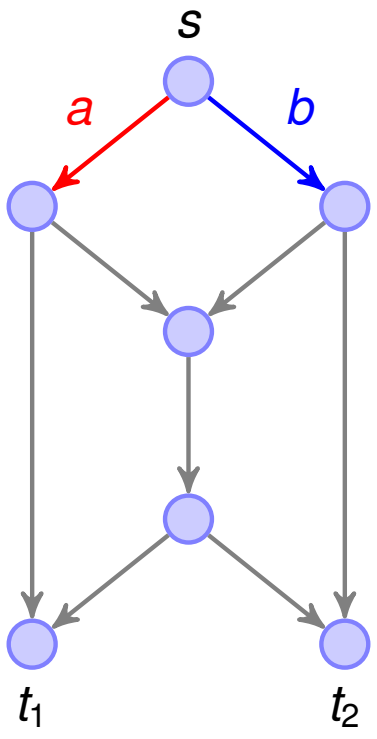
- C'est un graphe orienté, muni d'un ensemble S de sommets sources et d'un ensemble T de sommets destinataires.
- Les arêtes modélisent des canaux de communication qui transmettent sans erreurs des symboles d'un alphabet A.
- Chaque arête e peut transmettre c_e symboles par unité de temps (sa capacité).
- Objectif: maximiser la quantité d'information (nombre de symboles de A) transmise des sources aux destinataires.

Exemple introductif: le réseau papillon



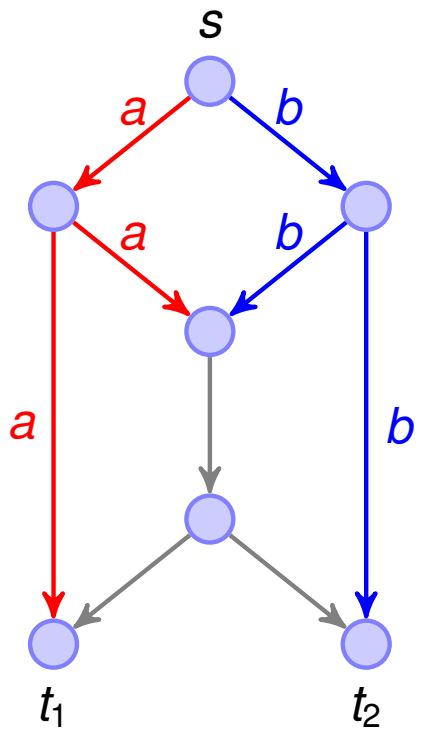
Une source, deux destinataires, capacité des arêtes 1.

Routage versus codage



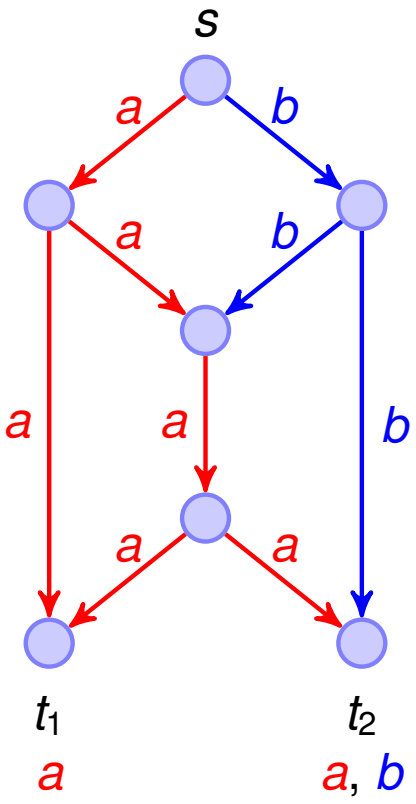
Routage: un noeud peut recopier le message reçu avant de le transférer.

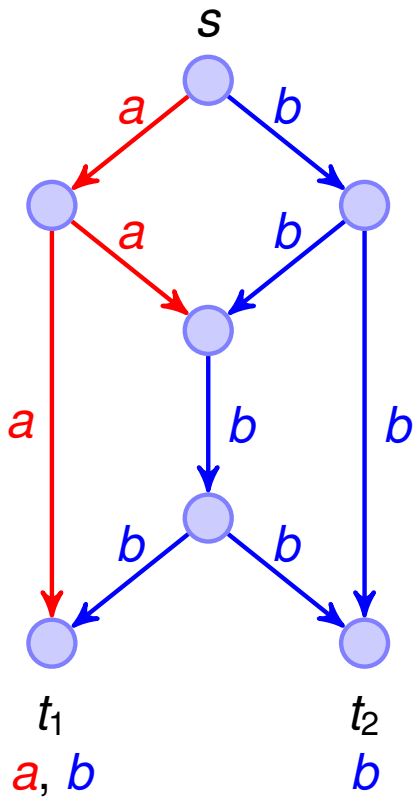
Routage versus codage



Le noeud du milieu reçoit *a* et *b* mais il doit choisir lequel transférer.

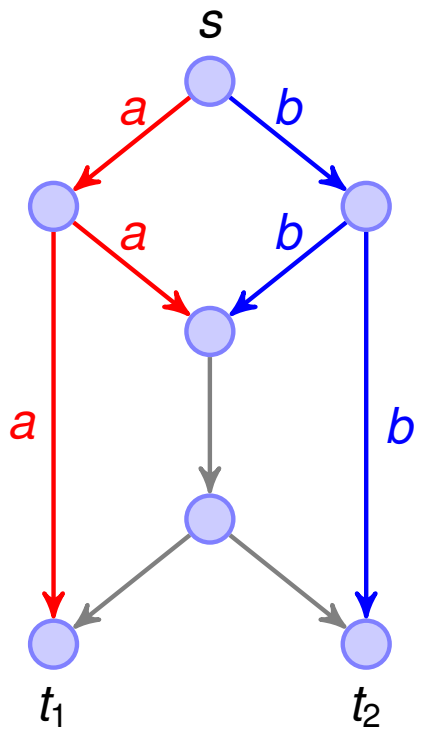
Routage versus codage





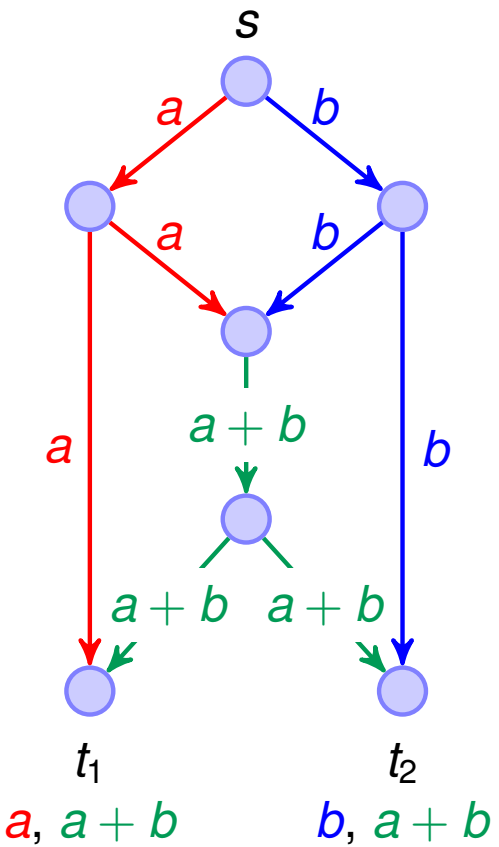
Dans les deux cas, on atteint un taux de transmission de 1.5.

Routage versus codage



Codage: les noeuds sont autorisés à *effectuer des opérations* sur les messages.

Routage versus codage



Le noeud du milieu choisit de transmettre $a + b$.

Chaque destinataire peut calculer a et b à partir des données qu'il reçoit. Le taux de transmission 2 est atteint!

Conclusions

- On peut transmettre deux symboles de la source à l'un quelconque des destinataires par routage en une session de communication.
- Si on veut transmettre ces deux symboles à **tous** les destinataires en une seule session, le routage ne marche pas, par contre le codage réussit.
- On va voir qu'en général:

  ★ Il y a une borne naturelle pour le taux de transmission donnée par le théorème max flot-min cut pour le flot d'un graphe.

  ★ Cette borne est atteinte par routage s'il y a un seul destinataire.

  ★ Cette borne est atteinte par codage linéaire s'il y a plusieurs destinataires et que l'alphabet est assez grand.

Plan

1. La notion de flot sur un graphe.
2. Le theorème max flot-min cut et le théorème de Menger.
3. Conséquence pour le taux de transmission dans un réseau de communication.
4. Optimalité du codage linéaire de réseau.

Références:

R. Ahlswede, N. Cai, S.-Y. R. Li, R. W. Yeung, *Network Information Flow*, IEEE Trans. Inf. Th., 2000

R. W. Yeung, *Information Theory and Network Coding*, Springer, 2008

Flot

- $G = (V, E)$ graphe orienté, $S = \{s\}$, $T = \{t\}$.
- Pour $v \in V$, on note $\mathrm{In}(v)$ et $\mathrm{Out}(v)$ l'ensemble des arêtes arrivant en v, respectivement partant de v.
- Pour $A \subset V$, $B \subset V$, on note $e : A \to B$ pour une arête commençant dans A terminant dans B.
- On note aussi $e' \to e$ pour deux arêtes consécutives.
- A chaque arête $e \in E$ est associé un nombre réel positif c_e (sa capacité).

Flot

Un flot est une application $f : E \to \mathbb{R}$ telle que:

- Pour tout $e \in E$, $0 \leq f(e) \leq c_e$.
- Pour tout $v \in V \setminus \{s, t\}$,

$$\sum_{e \in \text{In}(v)} f(e) = \sum_{e \in \text{Out}(v)} f(e)$$

C'est la loi de conservation du flot.

La valeur du flot f est définie par:

$$V(f) := \sum_{e \in \text{Out}(s)} f(e) - \sum_{e \in \text{In}(s)} f(e).$$

Modélise les réseaux de transports (véhicules, marchandise, fluides, etc..).

Pour $x \in V$, notons

$$f(x) := \sum_{e \in \mathrm{Out}(x)} f(e) - \sum_{e \in \mathrm{In}(x)} f(e).$$

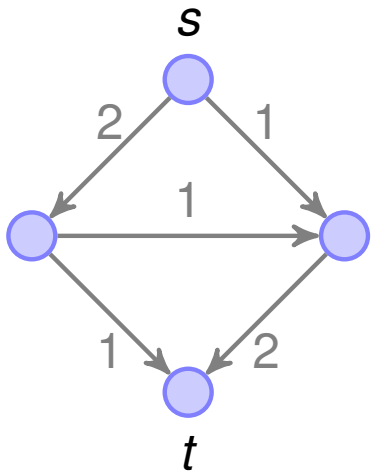
On remarque que:

$$V(f) = f(s) = -f(t).$$

En effet,

$$\sum_{e \in E} f(e) = \sum_{v \in V} \Big(\sum_{e \in \mathrm{Out}(v)} f(e) \Big) = \sum_{v \in V} \Big(\sum_{e \in \mathrm{In}(v)} f(e) \Big).$$

On obtient l'égalité annoncée en tenant compte de la loi de conservation : $f(x) = 0$ pour tout $x \neq s, t$.



$V(f) = 3$

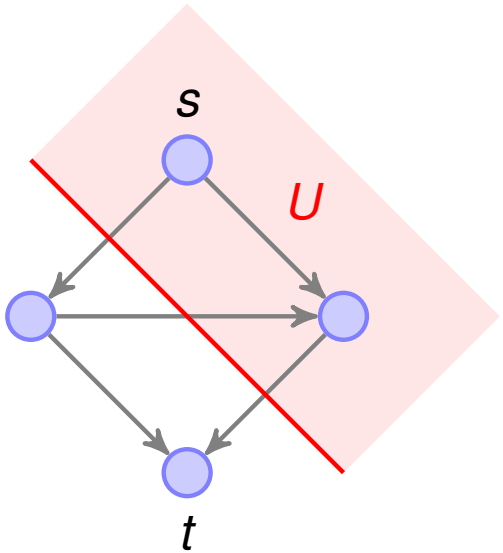
Coupe

Une coupe du graphe G séparant s et t est un ensemble U de sommets tel que $s \in U$ et $t \in \overline{U}$ (ensemble complémentaire de U).

La valeur de la coupe est

$$C(U) = \sum_{e:U \to \overline{U}} c_e$$

Exemple: si $c_e = 2$ pour tout $e \in E$, $C(U) = 4$.



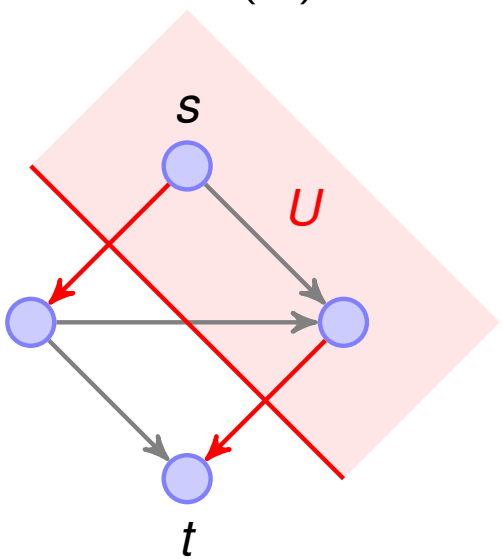
Coupe

Une coupe du graphe G séparant s et t est un ensemble U de sommets tel que $s \in U$ et $t \in \overline{U}$ (ensemble complémentaire de U).

La valeur de la coupe est

$$C(U) = \sum_{e:U \to \overline{U}} c_e$$

Exemple: si $c_e = 2$ pour tout $e \in E$, $C(U) = 4$.

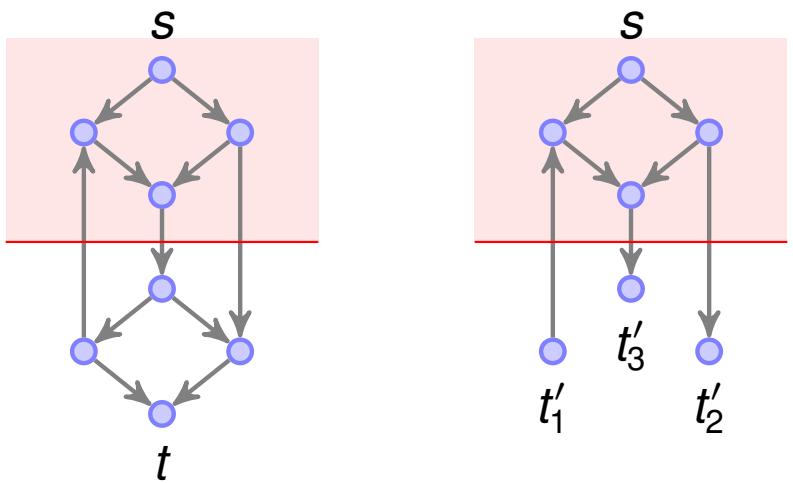


Le théorème max flot-min cut

Théorème: $\max_f V(f) = \min_U C(U)$.

Preuve: constructive.

Étape 1: pour tout flot f, coupe U, $V(f) \leq C(U)$: Soit T' l'ensemble des extrémités des arêtes coupées par U. On restreint le graphe à $U \cup T'$ et aux arêtes non contenues dans $\overline{U}$. Cela définit un nouveau graphe G'.



Ici $T' = \{t'_1, t'_2, t'_3\}$. La restriction de f est un flot sur G' de s à T' (la loi de conservation est respectée aux autres sommets).

Rappel: pour $x \in V$,

$$f(x) := \sum_{e \in \mathrm{Out}(x)} f(e) - \sum_{e \in \mathrm{In}(x)} f(e).$$

En notant, pour $X \subset V$, $f(X) = \sum_{x \in X} f(x)$, on a:

$$\begin{aligned} V(f) = f(s) = -f(T') &= \sum_{e : U \to \overline{U}} f(e) - \sum_{e : \overline{U} \to U} f(e) \\ &\leq \sum_{e : U \to \overline{U}} f(e) \leq \sum_{e : U \to \overline{U}} c_e = C(U). \end{aligned}$$

On remarque que l'égalité $V(f) = C(U)$ a lieu si et seulement si :

- $f(e) = c_e$ pour tout $e : U \to \overline{U}$
- $f(e) = 0$ pour tout $e : \overline{U} \to U$.

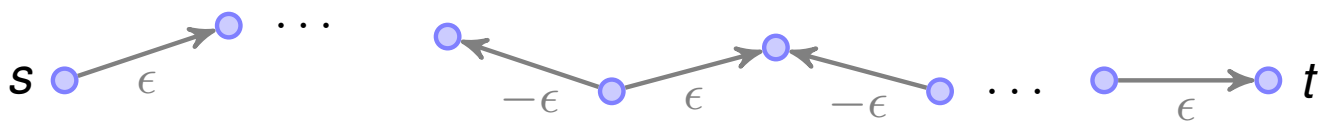
Étape 2: Il existe f, U tels que $V(f) = C(U)$: Soit f un flot, soit

U l'ensemble des extrémités de chemins sous-optimaux

$x \in U$ s'il existe $x_1 = s, x_2, \ldots, x_n = x$ tels que:

- $x_i x_{i+1}$ ou $x_{i+1} x_i$ appartient à E
- si $x_i x_{i+1} = e \in E$, $f(e) < c_e$
- si $x_{i+1} x_i = e \in E$, $f(e) > 0$.

Alors $s \in U$, et si $t \in U$, on peut augmenter f: en effet, dans ce cas, il existe un chemin sous-optimal de s à t.



Soit $\epsilon > 0$ tel que, le long de ce chemin,

$f(e) \leq c_e - \epsilon$ si e pointe vers t, $f(e) \geq \epsilon$ sinon.

On prend ϵ maximal, et on modifie f le long de ce chemin en posant:

$$f'(e) = f(e) \pm \epsilon$$

suivant la direction de e. Alors, f' est un flot, $V(f') = V(f) + \epsilon$ et le chemin n'est plus sous-optimal.

En itérant ce procédé, on obtient un flot tel que $t \notin U$. Alors U est une coupe séparant s et t, et, par construction,

- si $e : U \to \overline{U}$, $f(e) = c_e$
- si $e : \overline{U} \to U$, $f(e) = 0$.

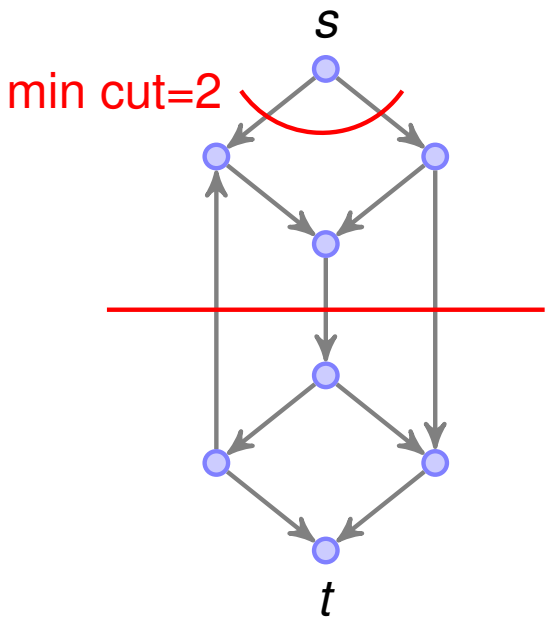
Donc, $V(f) = C(U)$, le flot est maximal et la coupe est minimale.

Remarques:

- La démonstration décrit un algorithme permettant de construire un flot maximal à partir du flot nul.
- La démonstration montre que, si les capacités c_e sont entières, il existe un flot maximal à valeurs entières.

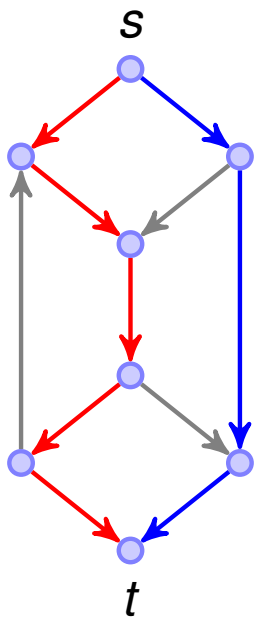
Le théorème de Menger

Théorème: [Menger, 1927] Si w est la valeur de la coupe minimale d'un graphe orienté dont les arêtes ont pour capacité 1, alors il existe w chemins orientés de s à t, deux à deux sans arêtes communes.



Le théorème de Menger

Théorème: [Menger, 1927] Si w est la valeur de la coupe minimale d'un graphe orienté dont les arêtes ont pour capacité 1, alors il existe w chemins orientés de s à t, deux à deux sans arêtes communes.



Preuve:

- Si on a n tels chemins, alors on peut définir un flot qui vaut 1 sur ces chemins et 0 ailleurs. Sa valeur est n donc par le théorème mf-mc, $n \leq w$.
- On en déduit le cas $w = 0$, puis on procède par récurrence sur w.
- Soit U l'ensemble des extrémités de chemins orientés commençant en s. On a $C(U) = 0$, donc $t \in U$ donc un chemin orienté relie s à t.
- Par l'algorithme, il existe un flot maximal $f_{\max}$ qui vaut 1 le long de ce chemin.
- On considère alors le graphe G' obtenu en enlevant les arêtes de ce chemin et le flot obtenu par restriction. Sa valeur est $V(f_{\max}) - 1 = w - 1$ donc G' a pour valeur de coupe minimale $w - 1$ donc contient par récurrence $w - 1$ chemins orientés disjoints.

Codage de réseau

Informellement:

- $G = (V, E)$ graphe orienté acyclique, $S = \{s\}$, $T = \{t_1, \ldots, t_\ell\}$
- Les arêtes transmettent des éléments d'un alphabet A.
- À l'instant 0, $X \in A^w$ est émis par s
- À chaque instant $k = 1, 2, \ldots, K$, un élément de A est transmis sur une arête $e(k)$, qui est une fonction des éléments transmis sur les arêtes entrantes aux temps précédents.
- Chaque arête transmet au plus un nombre c_e d'élément de A au cours de la session.
- À la fin, à chaque t_i, une fonction de décodage D_i appliquée aux éléments arrivés en t_i retourne un élément Y_i de A^w.

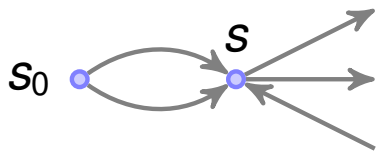
Si, pour tout $i = 1, \ldots, \ell$, D_i retourne X, on dit que le taux de transmission multicast de ce schéma de codage est w.

Codage de réseau

Sans perte de généralité:

- On peut supposer $c_e = 1$:

- On peut supposer $\text{In}(s) = \emptyset$ et $\text{Out}(t_i) = \emptyset$:



Codage de réseau

Plus formellement:

- Une application: (l'ordre des transmissions)

$$\{1, \ldots, K\} \to E$$
$$k \mapsto e(k)$$

  telle que, pour tout $e \in E$, $e(k) = e$ au plus une fois.

- Des applications: ($f_k(X)$ est transmis sur $e(k)$)

$$A^w \to A$$
$$X \mapsto f_k(X)$$

  telles que: il existe des applications φ_k avec

$$f_k(X) = \varphi_k(f_{k'}(X) : k' < k, e(k') \to e(k)) \text{ si } e(k) \notin \mathrm{Out}(s)$$

- Pour $i = 1, \ldots, \ell$, des applications $D_i : A^{|\ln(t_i)|} \to A^w$.

Codage de réseau

- Le taux de transmission multicast est w si, pour tout $X \in A^w$,

$$\text{Pour tout } i = 1, \ldots, \ell,\ D_i(f_k(X) : e(k) \in \text{In}(t_i)) = X.$$

On supposera que le graphe G est acyclique car dans ce cas:

- Il existe un ordre sur les arêtes 'de haut en bas', i.e. tel que:

$$\text{Si } e' \to e \text{ alors } e' < e.$$

- On prendra toujours un tel ordre pour les transmissions pour ne pas avoir de problème de 'délai' (toute l'information est arrivée à un noeud *avant* que celui-ci transfère).
- Il suffit alors de spécifier les *fonctions globales* $x_e := f_e(X)$ ou les *fonctions locales* $\varphi_e(x_{e'} : e' \to e)$.

Théorème: Si le taux de transmission multicast est w alors, pour tout $i = 1, \ldots, \ell$,

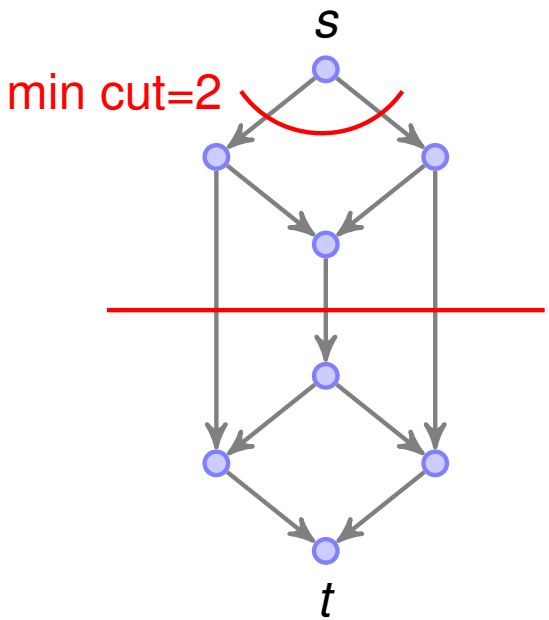
$$w \leq \min \text{ cut}(G, s, t_i).$$

Preuve: Facile. Si U est une coupe séparant s et t_i, les données qui arrivent en t_i sont fonction des données qui transitent par les arêtes $e : U \to \overline{U}$. Plus un peu de théorie de l'information.

Le routage est optimal avec un unique destinataire

Théorème: Si $T = \{t\}$, le taux de transmission $w = \min \text{cut}(G)$ est atteint par routage.

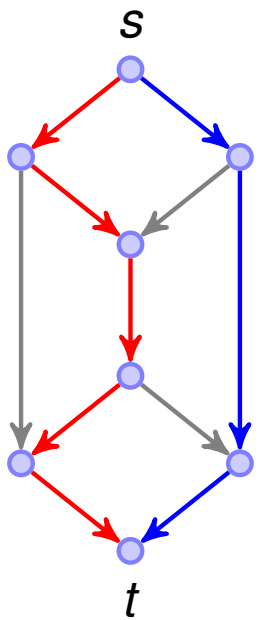
Preuve: Par le théorème de Menger, il existe $w = \min \text{cut}(G)$ chemins orientés sans arête commune de s à t. On peut transmettre le long de ces chemins w symboles distincts.



Le routage est optimal avec un unique destinataire

Théorème: Si $T = \{t\}$, le taux de transmission $w = \min \text{cut}(G)$ est atteint par routage.

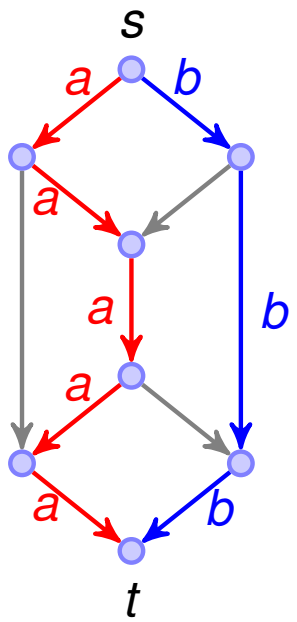
Preuve: Par le théorème de Menger, il existe $w = \min \text{cut}(G)$ chemins orientés sans arête commune de s à t. On peut transmettre le long de ces chemins w symboles distincts.



Le routage est optimal avec un unique destinataire

Théorème: Si $T = \{t\}$, le taux de transmission $w = \min \text{cut}(G)$ est atteint par routage.

Preuve: Par le théorème de Menger, il existe $w = \min \text{cut}(G)$ chemins orientés sans arête commune de s à t. On peut transmettre le long de ces chemins w symboles distincts.



Le codage linéaire est optimal avec plusieurs destinataires

Théorème: [Li, Yeung, Cai, 2003] Si $A = \mathbb{F}_q^m$ et $\mathbb{F}_q$ est un corps fini assez grand, le codage linéaire permet d'atteindre le taux de transmission multicast

$$w = \min_{i=1,\ldots,\ell} \min \text{ cut}(G, s, t_i).$$

Codage linéaire: les applications $f_k(X)$ sont $\mathbb{F}_q$-linéaires.

Koetter et Médard, 2003: preuve algébrique ($q > |T|$).

Jaggi, Sanders, 2005: Linear Information Flow Algorithm ($q \geq |T|$).

Notation: $X \in A^w = (\mathbb{F}_q^m)^w$ est identifié à la matrice $X \in \mathbb{F}_q^{w \times m}$ dont les lignes sont les 'paquets' $X_i \in \mathbb{F}_q^m$.

Preuve algébrique

Pour tout $i = 1, \ldots, \ell$, la matrice Y_i reçue en t_i s'écrit

$$Y_i = T_i X, \ T_i \in \mathbb{F}_q^{w \times w}$$

Ce que l'on veut: pour tout $i = 1, \ldots, \ell$, T_i inversible soit

$$\det(T_1) \ldots \det(T_\ell) \neq 0.$$

Soit $x_e \in \mathbb{F}_q^m$ transmis par l'arête e au cours de la session. À chaque paire d'arêtes consécutives (e', e) est associé un coefficient $\lambda_{e',e} \in \mathbb{F}_q$ tel que:

$$x_e = \sum_{e' \to e} \lambda_{e',e} x_{e'}.$$

En itérant, on voit que les coefficients des matrices T_i sont des polynômes en les $\lambda_{e',e}$. Donc $\prod_{i=1}^{\ell} \det(T_i)$ est un polynôme en les variables $\lambda_{e',e}$. Donc, si $|\mathbb{F}_q|$ est assez grand, on peut choisir des valeurs pour $\lambda_{e',e}$ de telle sorte qu'il ne s'annule pas.

L'algorithme de Jaggi-Sanders

On va calculer itérativement des vecteurs $F_e \in \mathbb{F}_q^w$ tels que

$$x_e = F_e X.$$

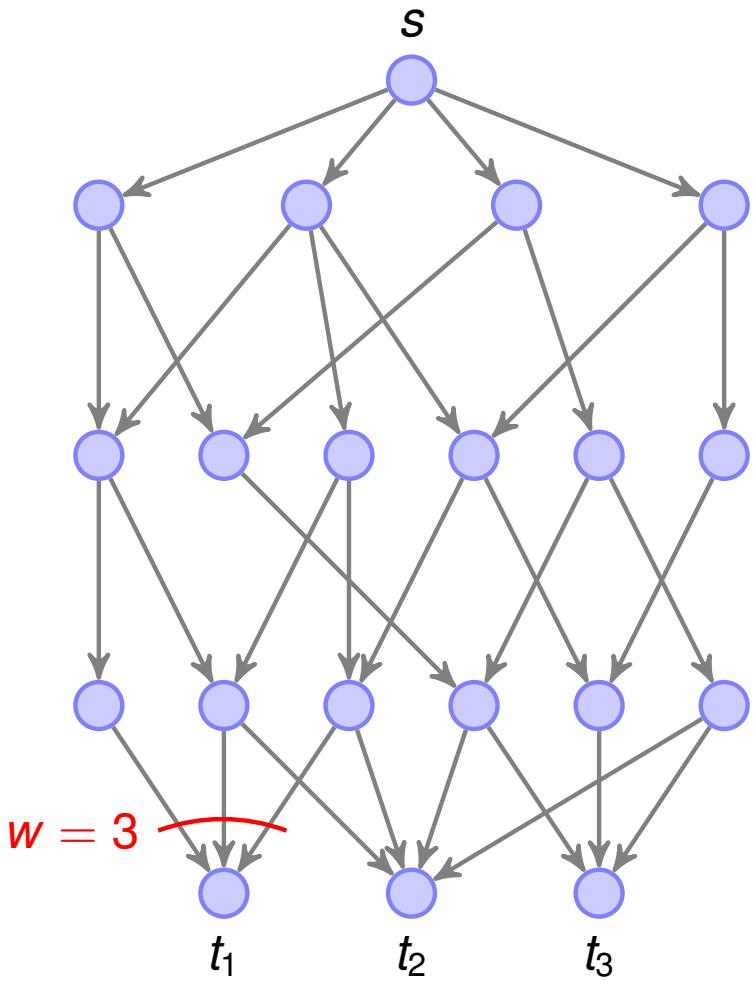
Ce qu'on veut: pour tout $i = 1, \ldots, \ell$,

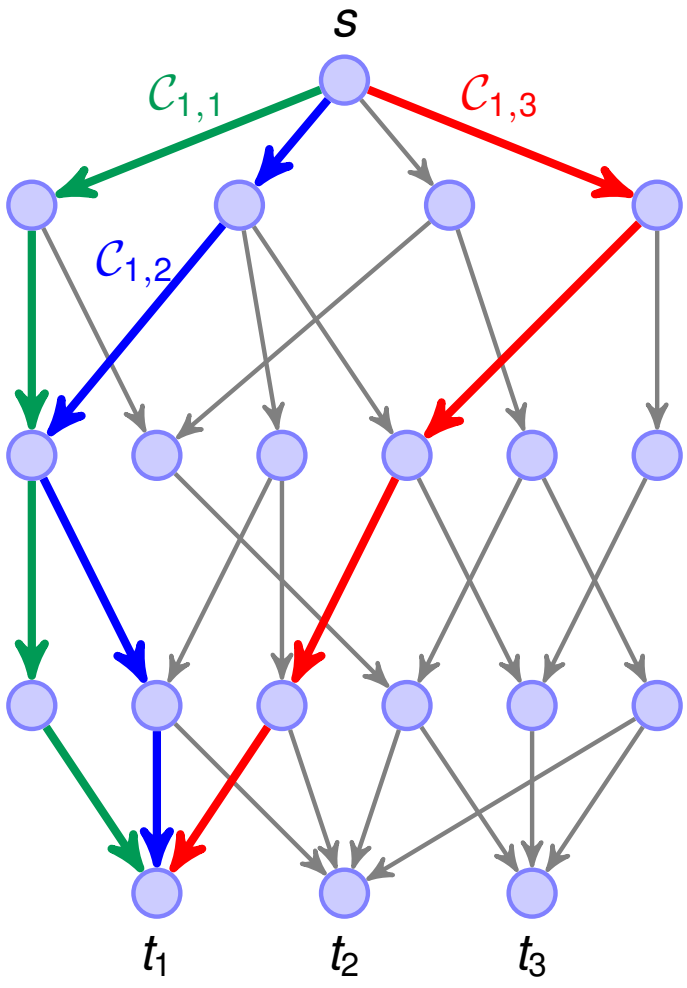
$$\{F_e \ : \ e \in \mathrm{In}(t_i)\} \text{ engendre } \mathbb{F}_q^w.$$

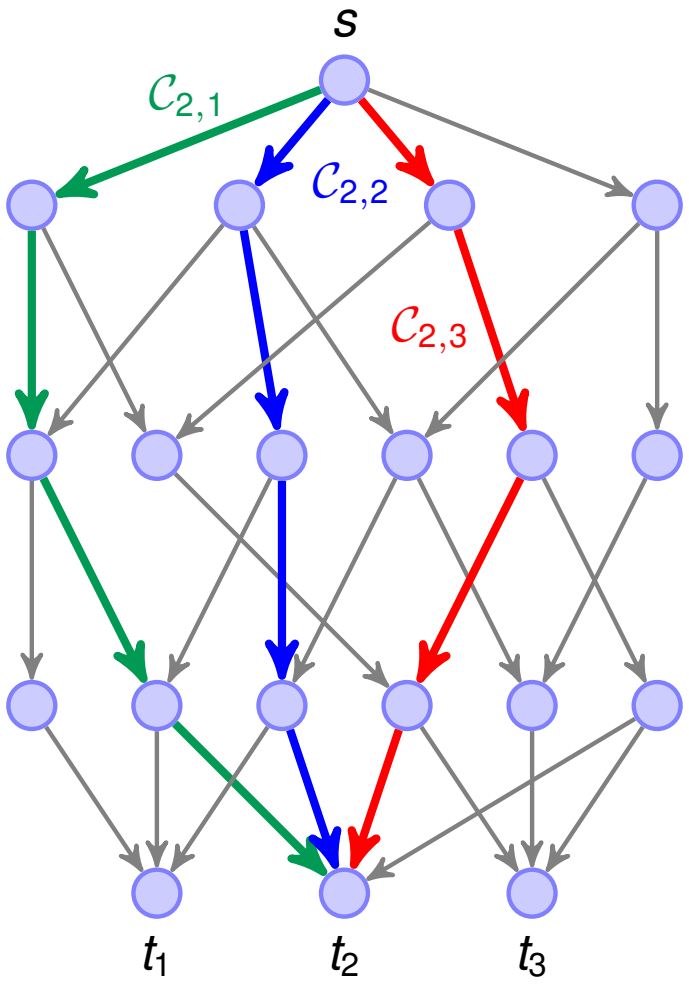
Pour chaque $i = 1, \ldots, \ell$, par le Th de Menger, il existe w chemins orientés disjoints joignant s à t_i. Notons

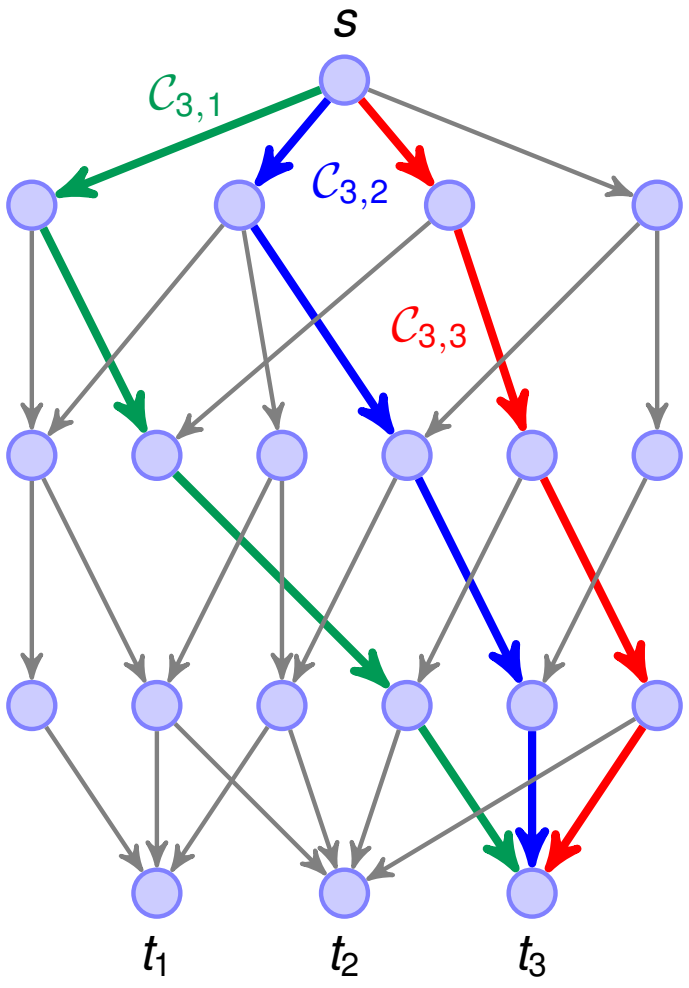
$$\mathcal{P}_i = \{\mathcal{C}_{i,1}, \ldots, \mathcal{C}_{i,w}\}$$

On va calculer dans un ordre 'de haut en bas' les F_e de sorte que le rang soit maximal le long de chacun des $\mathcal{P}_i$.

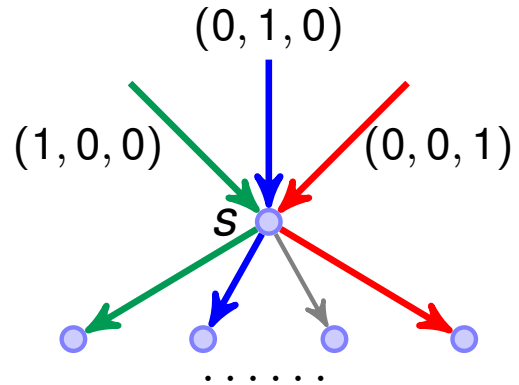

s
w = 3
t1
t2
t3


s
$C_{1,1}$
$C_{1,2}$
$C_{1,3}$
t_1
t_2
t_3


s
$C_{2,1}$
$C_{2,2}$
$C_{2,3}$
t_1
t_2
t_3


s
$\mathcal{C}_{3,1}$
$\mathcal{C}_{3,2}$
$\mathcal{C}_{3,3}$
t_1
t_2
t_3

- On élimine toutes les arêtes n'appartenant pas à l'un des $\mathcal{C}_{i,j}$.
- On rajoute w arêtes en amont de s et on initialise leurs F_e avec la base canonique de $\mathbb{F}_q^w$.

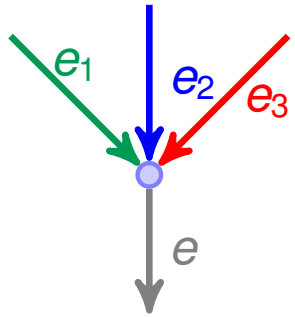


- On parcourt les arêtes suivant un ordre 'de haut en bas' pour choisir F_e.
- Soit, sur chaque chemin $\mathcal{C}_{i,j}$, $F_{i,j}$ le dernier des F_e calculé. Soit

$$W_i = \mathrm{Vec}(F_{i,j} \ : \ j = 1, \dots w).$$

À l'initialisation, on a $\dim(W_i) = w$. Notre objectif est de *conserver le rang w dans tous les W_i*.

- Supposons $e \in \mathcal{C}_{i_1,j_1}, \ldots, \mathcal{C}_{i_s,j_s}$. Notons e_r son prédécesseur dans $\mathcal{C}_{i_r,j_r}$.



  Notons que les indices i_r sont deux à deux distincts donc $s \leq \ell$.

- Soit

$$W = \mathrm{Vec}(F_{e_1}, \ldots, F_{e_s}) \quad (e_r \in \mathcal{C}_{i_r,j_r}).$$

  et soit

$$W'_{i_r} = \mathrm{Vec}(F_{i_r,j} \ : \ j \neq j_r).$$

  On va choisir F_e tel que:

$$F_e \in W \text{ (ainsi } x_e \text{ est comb lin des } x_{e'},\ e' \to e)$$
$$F_e \notin W'_{i_r} \text{ (ainsi on maintient le rang } w \text{ dans } W_{i_r})$$

- On a : $\dim(W'_{i_r} + W) \geq \dim(W'_{i_r}) + 1$ donc

$$\dim(W'_{i_r} \cap W) = \dim(W'_{i_r}) + \dim(W) - \dim(W'_{i_r} + W) \leq \dim(W) - 1.$$

- Donc, si $m = \dim(W)$,

$$\text{card}\left(\bigcup_{r=1}^{s}(W'_{i_r} \cap W)\right) \leq \ell(q^{m-1} - 1) + 1.$$

- Si $q \geq \ell$, on a $q^m > \ell(q^{m-1} - 1) + 1$ donc

$$\text{il existe } F_e \in W \setminus \bigcup_{r=1}^{s}(W'_{i_r} \cap W).$$

- Si on choisit F_e aléatoirement dans W,

$$\text{proba d'échec} \leq \frac{\ell(q^{m-1}-1)+1}{q^m} \leq \frac{\ell}{q}$$

donc

$$\text{proba de succès } \geq 1-\frac{\ell}{q}.$$

- Si chaque arête e choisit F_e aléatoirement, on parle de codage linéaire aléatoire. La probabilité qu'un tel schéma atteigne le taux de transmission optimal est donc au moins

$$\left(1-\frac{\ell}{q}\right)^{|E|} \simeq 1-\frac{\ell|E|}{q} = 1-o(1).$$

Conclusions

- Avec une source et un destinataire, le routage permet d'atteindre la limite théorique min cut(G) par les chemins prévus par le th de Menger.
- Avec une source et ℓ destinataires, le routage ne permet pas d'atteindre la limite théorique $\min_{i=1}^{\ell}$ min cut(G, s, t_i).
- Par contre, dès que $|A| \geq \ell$, le codage linéaire le permet (algo Jaggi-Sanders).
- Si $|A| \gg \ell$, le codage linéaire aléatoire réussit avec probabilité proche de 1. Avantage: pas de connaissance à priori du graphe.
- Ces résultats s'étendent aux graphes contenant des cycles. Par contre, le cas des réseaux multi-source est **beaucoup** moins bien compris..