Estimation des Matrices de Trafic Origine Destination

Sandrine VATON

ENST Bretagne, département Informatique

e-mail: Sandrine.Vaton@enst-bretagne.fr

Personnes participant aux travaux autour de la métrologie à l'ENST Bretagne, département Informatique (Brest)

- Annie Gravey, Directeur de Recherche, responsable du département Informatique
- Alain Leroy, Ingénieur d'Études,
  - enseignements : Unix, Réseaux, Voix sur IP, Qualité de Service
  - encadre le stage de Sergio Morant sur la mesure de trafic avec Ethereal
- Sandrine Vaton, Maître de Conférences,
  - enseignements : codage correcteur d'erreurs, théorie de l'information, méthodes de Monte Carlo par Chaînes de Markov, cryptographie
  - thèse sur la modélisation de trafic, décembre 1998, ENST département TSI
- Sergio Morant, stagiaire, mesure de trafic avec Ethereal

Matrice de trafic Origine Destination :

- c'est la quantité de trafic qui part de *tel* noeud Origine dans le réseau pour aller vers *tel* noeud Destination ; elle est représentée sous la forme d'une matrice à 2 entrées
- utile pour la conception, le dimensionnement, l'ingénierie de trafic, la tarification...

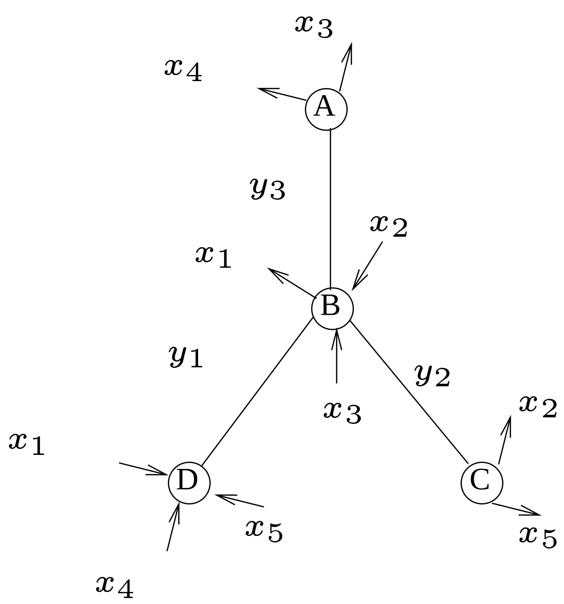
Exemples d'utilisation de la matrice de trafic :

- *Administrateur d'un réseau local :*
  - constate qu'un lien est particulièrement chargé,
  - soit augmente la capacité de ce lien,
  - soit modifie les tables de routage pour rééquilibrer le trafic $\Rightarrow$ la matrice de trafic peut être utile pour identifier les flots les plus importants
- *Fournisseur d'accès Internet :*
  - peut utiliser la matrice de trafic pour identifier ses plus gros clients et les charger en conséquence

Problématique :

- une Origine (ou une Destination) est l'ensemble des machines atteintes à partir d'un port d'un routeur en bord du réseau (c'est un groupe d'adresses IP, un réseau local par exemple) ;
- volume de trafic sur chaque lien interne du réseau :
  - ces données sont faciles à obtenir à l'aide du protocole SNMP, supporté par l'ensemble des routeurs commerciaux ;
  - le protocole SNMP fournit des statistiques sur le volume de trafic entrant et sortant pour chacun des ports des routeurs ;
  - l'utilisation des données SNMP des routeurs du coeur du réseau permet d'obtenir le volume de trafic sur chacun des liens du réseau.
  - les données SNMP sont utilisées pour tous les administrateurs de réseaux.
- **Problème :**
  - sur chaque lien du réseau transitent des sources de trafic ayant des Origines et des Destinations variées ;
  - on n'observe que le trafic multiplexé au niveau des liens ;
  - comment en déduire la valeur des flots OD ? $\Rightarrow$ **la matrice de trafic**

Exemple : un réseau avec 4 noeuds $\{A, B, C, D\}$, 3 liens monodirectionnels $\{y_1, y_2, y_3\}$ et 5 couples Origine Destination (OD) $\{x_1, x_2, x_3, x_4, x_5\}$



La structure de routage est connue et, dans cet exemple, déterministe :

$$x_1 \ : D \rightarrow B \quad x_2 \ : B \rightarrow C \quad x_3 \ : B \rightarrow A \quad x_4 \ : D \rightarrow B \rightarrow A \quad x_5 \ : D \rightarrow B \rightarrow C$$

De façon équivalente, le trafic au niveau de chaque lien résulte du multiplexage de différents flots OD :

$$y_1 = x_1 + x_4 + x_5 \quad y_2 = x_2 + x_5 \quad y_3 = x_3 + x_4$$

Ce système linéaire peut aussi s'écrire sous la forme matricielle :

$$\mathbf{y} = \mathbf{A}\mathbf{x} \quad \mathbf{x} = (x_1, x_2, x_3, x_4, x_5)^T \quad \mathbf{y} = (y_1, y_2, y_3)^T$$

avec **A** la matrice de routage qui vaut, dans cet exemple, $\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}$.

Matrice de trafic Origine Destination

La matrice de trafic Origine Destination (OD) est la quantité $\mathbf{x}$ (représentée ici sous la forme d'un vecteur colonne par commodité).

Dest. **Orig.**	A	B	C	D
A	0	0	0	0
B	x_3	0	x_2	0
C	0	0	0	0
D	x_4	x_1	x_5	0

Problème :

- on mesure le débit au niveau de chacun des liens du réseau (protocole SNMP,...) $\Rightarrow \mathbf{y}$
- on désire en déduire la valeur de chacun des flots Origine Destination $\Rightarrow \mathbf{x}$

$\Rightarrow$ estimer $\mathbf{x}$ sachant $\mathbf{y}$ sous la contrainte $\mathbf{y} = \mathbf{A}\mathbf{x}$ et sous des contraintes de positivité sur les flots OD $x_i \geq 0, i = 1, 5$.

Exemple :

Supposons que $x_1 = 10$, $x_2 = 2$, $x_3 = 18$, $x_4 = 15$, $x_5 = 50$.

$$\text{Alors,} \begin{cases} y_1 = x_1 + x_4 + x_5 = 75, \\ y_2 = x_2 + x_5 = 52, \\ y_3 = x_3 + x_4 = 33. \end{cases}$$

$\Rightarrow$ $\mathbf{x} = (10, 2, 18, 15, 50)$ est solution de $\mathbf{y} = \mathbf{A}\mathbf{x}$ mais **x'** $= (53, 50, 13, 20, 2)$ est également solution, de même que **x"** $= (0, 2, 8, 25, 50)$... et les 3 solutions n'ont pas grand chose à voir entre elles !

$\hookrightarrow$ il y a une infinité de solutions réelles (l'espace des solutions est de dim. 2 car la matrice $\mathbf{A}$ est de rang 3) et un grand nombre de solutions entières positives.

Contraignons un peu plus le problème : supposons maintenant que l'on sache, *a priori*, que x_1, x_2, x_3, x_4 et x_5 sont distribuées selon des lois de Poisson de moyenne respectivement 10, 1, 20, 10, 50 :

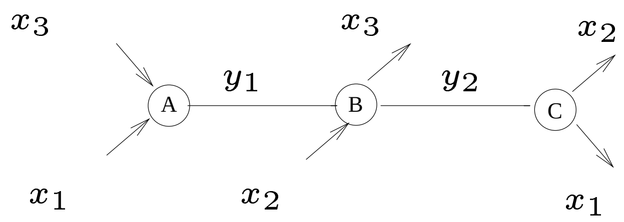
$$x_1 \sim \mathcal{P}(10) \quad x_2 \sim \mathcal{P}(1) \quad x_3 \sim \mathcal{P}(20) \quad x_4 \sim \mathcal{P}(10) \quad x_5 \sim \mathcal{P}(50)$$

Alors la solution $\mathbf{x} = (10, 2, 18, 15, 50)$ est nettement plus vraisemblable que la solution $\mathbf{x'} = (53, 50, 13, 20, 2)$ ou que la solution $\mathbf{x''} = (0, 2, 8, 25, 50)$.

$\hookrightarrow$ L'introduction d'une loi a priori fait apparaître certaines solutions comme beaucoup plus vraisemblables que les autres !

$\hookrightarrow$ Mais ... le résultat de l'estimation est très fortement biaisé par le choix de la loi a priori.

Un exemple simple : 2 liens monodirectionnels, 3 couples OD



Supposons qu'on ait mesuré $y_1 = x_1 + x_3 = 3$ et $y_2 = x_1 + x_2 = 5$.

Loi A Priori $\lambda = (2,2,2)$		Loi A Priori $\lambda = (1,3,2)$	
(x_1, x_2, x_3)	vraisemblance $p(\mathbf{x};\lambda)$	(x_1, x_2, x_3)	vraisemblance $p(\mathbf{x};\lambda)$
(0,5,3)	0,0009	(0,5,3)	0,0067
(1,4,2)	0,0066	**(1,4,2)**	**0,0167**
(2,3,1)	**0,0132**	(2,3,1)	0,0112
(3,2,0)	0,0066	(3,2,0)	0,0019

$\hookrightarrow$ 5 solutions entières possibles

$\hookrightarrow$ la solution la plus vraisemblable est $(x_1 = 2, x_2 = 3, x_3 = 1)$ si $\lambda = (2,2,2)$ et $(x_1 = 1, x_2 = 4, x_3 = 2)$ si $\lambda = (1,3,2)$

$\Rightarrow$ la loi a priori ne doit pas être choisie n'importe comment, elle doit être **estimée** !

- Quel modèle (Poisson, mélange, Markov, fractal...) ?
- Quel modèle pour quelle granularité de temps ?
- Comment estimer les paramètres de ce modèle ?
- Faut-il prendre en compte les non stationnarités du trafic ?
- ...

Hypothèses standards

- Généralement on suppose qu'on fait des mesures réitérées sur le réseau. Exemple : statistiques SNMP sur tous les routeurs du réseau relevées toutes les 20 secondes pendant 1 heure.
- Généralement, modèle statistique pour les flots OD :
  - soit Poissonien, soit Gaussien,
  - les valeurs successives pour une même paire OD sont indépendantes les unes des autres (contradictoire avec les modèles markoviens, la dépendance longue,...),
  - les valeurs successives pour une même paire OD ont même distribution (hypothèse de stationnarité)
- Routage déterministe / routage probabiliste.

Antériorités

- Travaux sur l'estimation de la matrice de trafic dans les réseaux de communication. Utilisée par les concepteurs de réseaux (routiers,...) pour tracer des voies de communication directes pour les axes les plus fréquentés.
- Travaux sur l'estimation de la matrice de trafic en téléphonie commutée.
- Y.Vardi, *Network Tomography : Estimating Source Destination Traffic Intensities From Link Data*, JASA, 1996
  - mesures réitérées sur le réseau : $\mathbf{y}^{(k)}, k = 1, K$;
  - hypothèse poissonnienne sur les flots OD : $x_i^{(k)} \sim \mathcal{P}(\lambda_i)$;
  - estimation ML des λ de Poisson par une méthode EM :

$$\lambda^{n+1} = \frac{1}{K} \sum_{k=1,K} \mathbb{E}(\mathbf{x}^{(k)} \mid \mathbf{y}^{(k)}; \lambda^n)$$

    - l'étape E, $\mathbb{E}(\mathbf{x}^{(k)} \mid \mathbf{y}^{(k)}; \lambda)$, n'est pas calculable sous l'hypothèse de Poisson car il faudrait énumérer l'ensemble des $\mathbf{x}^{(k)}$ solutions de $\mathbf{y}^{(k)} = \mathbf{A}\mathbf{x}^{(k)}$;
    - approximation gaussienne pour simplifier l'étape E $\Rightarrow$ détermination de $\mathbb{E}(\mathbf{x}^{(k)} \mid \mathbf{y}^{(k)}; \lambda)$ par la droite de régression (gaussienne multivariée)

- cette approximation n'est valide pour des valeurs de λ_i élevées ;
- démontre la concavité locale de la fonction de vraisemblance $\Rightarrow$ viabilité de la méthode EM (pas de problèmes d'extrema locaux au voisinage de λ^*).

– Estimation MLE avec approximation gaussienne de $\bar{\mathbf{y}} = \frac{1}{K}\sum_{k=1,K}\mathbf{y}^{(k)} \sim \mathcal{N}(\mathbf{A}\lambda, \mathbf{A}\Lambda\mathbf{A}^T)$
$\Rightarrow$ conduit à un problème des moindres carrés pondérés, les pondérations étant calculées à partir des covariances empiriques des $\mathbf{y}^{(k)}$:

$$\hat{\lambda} = \operatorname{Arg}\min_{\lambda}(\bar{\mathbf{y}} - \mathbf{A}\lambda)^T(\hat{\mathrm{cov}}(\mathbf{y}))^{-1}(\bar{\mathbf{y}} - \mathbf{A}\lambda)^T$$

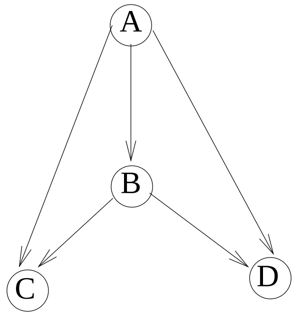
– Estimation basée sur les équations de moments d'ordre 1 et 2 :

$$\mathbb{E}(\mathbf{y}) = \mathbf{A}\lambda \qquad \mathrm{cov}(y_i, y_{i'}) = \textstyle\sum_j a_{ij}a_{i'j}\lambda_j$$

- mise en oeuvre de la minimisation par une méthode EM (méthode LININPOS)

- régularisation : les équations de moment d'ordre 2 sont pondérées par $\epsilon < 1$;
  - la forme des moments d'ordre 1 ne suppose pas l'hypothèse de Poisson
  - par contre l'expression des moments d'ordre 2 dépend très fortement de l'hypothèse de Poisson
  - plus ϵ est petit plus on s'éloigne du modèle poissonnien ;
- prise en compte des équations inconsistentes avec l'hypothèse poissonnienne :
  Exemple : $\sum_j a_{ij} a_{i'j} = 0$ et $\hat{cov}(y_i, y_{i'}) > 0$
  $\hookrightarrow$ on laisse tomber les équations inconsistentes, il y a suffisamment de moments pour se le permettre

– Routage probabiliste :



Matrice de routage **A**					
paire OD	AB	AC	AD	BC	BD
lien					
$A \to B$	1.0	0.2	0.3		
$A \to C$		0.8			
$A \to D$			0.7		
$B \to C$		1.0		1.0	
$B \to D$			1.0		1.0

$$y_i \sim \mathcal{P}(\sum_j P_i^j \lambda_j) \quad \mathbb{E}(\mathbf{y}) = \mathbf{P}\lambda \quad \mathrm{var}(\mathbf{y}) = \mathbf{P}\lambda \quad \mathrm{cov}(y_i, y_{i'}) = \sum_j P_{ii'}^j \lambda_j$$

avec P_i^j la probabilité qu'un paquet OD j passe par le lien i et avec $P_{ii'}^j$ la probabilité qu'un paquet OD j emprunte à la fois les liens i et i'.

$\hookrightarrow$ algorithme permettant de calculer les quantités P_i^j à partir de la matrice de routage

- C.Tebaldi, M.West, *Bayesian Inference of Network Traffic Using Link Count Data*, JASA 1998.
  - approche bayésienne
  - objectif : simulation de $\mathbf{x} = (x_1, \cdots, x_c)$ (les flots OD) sous leur loi jointe conditionnellement à $\mathbf{y} = (y_1, \cdots, y_r)$ (les liens)
  - méthode : méthode de Monte Carlo par Chaînes de Markov (MCMC), algorithme de Metropolis within Gibbs
    - échantillonnage de Gibbs : une itération de l'algorithme de Gibbs se décompose comme suit,

$$
\begin{array}{lcl}
x_1^{t+1} & \leftarrow & p_1(x_1 \mid x_2^t, \ldots, x_c^t, \mathbf{y}) \\
x_2^{t+1} & \leftarrow & p_2(x_2 \mid x_1^{t+1}, x_3^t, \ldots, x_c^t, \mathbf{y}) \\
\vdots & & \\
x_i^{t+1} & \leftarrow & p_i(x_i \mid x_1^{t+1}, \ldots, x_{i-1}^{t+1}, x_{i+1}^t, \ldots, x_c^t, \mathbf{y}) \\
\vdots & & \\
x_c^{t+1} & \leftarrow & p_c(x_c \mid x_1^{t+1}, \ldots, x_{c-1}^{t+1}, \mathbf{y})
\end{array}
$$

On réactualise “OD flot” par “OD flot”! Le vecteur $\mathbf{x}^t$ converge en loi vers $p(\mathbf{x} \mid \mathbf{y})$ quand t tend vers l'infini.

- Tebaldi établit l'expression de $p_i(x_i \mid \mathbf{x}_{-i}, \mathbf{y})$, la densité de probabilité de x_i sachant $\mathbf{x}_{-i}$ et $\mathbf{y}$, :
  - ce n'est pas une loi simple,
  - la densité de probabilité n'est connue qu'à un facteur multiplicatif près
- méthode de Metropolis pour simuler la loi $p_i(x_i \mid \mathbf{x}_{-i}, \mathbf{y})$
  $\rightarrow$ principe : recycler, par un mécanisme d'acceptation rejet, des échantillons produits selon une loi *instrumentale* pour en faire des échantillons produits selon la loi *objectif*, ici la loi $p_i(x_i \mid \mathbf{x}_{-i}, \mathbf{y})$;
  $\rightarrow$ la densité de la loi objectif $p_i(x_i \mid \mathbf{x}_{-i}, \mathbf{y})$ peut être définie à un facteur multiplicatif près, ce n'est pas un problème.

– article de C.Tebaldi et M.West commenté par Y.Vardi dans JASA 1998 ; principaux reproches :

- le résultat obtenu dépend très fortement de la loi a priori choisie pour $\mathbf{x}$ (Poisson, lambda=?)
- Tebaldi n'envisage pas le cas de mesures répétées sur le réseau
- λ n'est pas estimé mais fixé a priori

- J.Cao, D.Davis, S.Vander Wiel, B.Yu, *Time-Varying Network Tomography : Router Link Data*, JASA 2000
  - cette fois ci la matrice de trafic est supposée varier dans le temps
  - modèle local pour les valeurs successives des flots OD : indépendants et identiquement distribués

$$\begin{aligned} & \mathbf{x}^{(k-h)}, \cdots, \mathbf{x}^{k}, \cdots, \mathbf{x}^{(k+h)} \sim \text{iid } \mathcal{N}(\lambda^{(k)}, \Sigma^{(k)}) \\ \text{avec} \quad & \Sigma^{(k)} = \phi^{(k)} \text{diag}((\lambda^{(k)})^c), c = 1, 2 \end{aligned}$$

    $\hookrightarrow$ estimation de $\phi^{(k)}$ et de $\lambda^{(k)}$ au vu de $\mathbf{x}^{(k-h)}, \cdots, \mathbf{x}^{(k)}, \cdots, \mathbf{x}^{(k+h)}$
  - lissage supplémentaire en supposant que le vecteur des paramètres $\eta^{(k)} = (\log \lambda^{(k)}, \log \phi^{(k)})$ suit une marche aléatoire gaussienne :

$$\eta^{(k+1)} = \eta^{(k)} + v^{(k)} \qquad v^{(k)} \sim \mathcal{N}(0, V)$$

  - validation sur données réelles obtenues sur le réseau local de Lucent Technologies

- J.Cao, S.Vander Wiel, B.Yu, Z.Zhu, *A Scalable Method for Estimating Network Traffic Matrices from Link Counts*
  - problème : réduire la complexité numérique de l'estimation pour pouvoir traiter le cas des réseaux avec un grand nombre de routeurs
  - solution : décomposer le problème en plusieurs sous-problèmes
    - grouper les paires OD de telle façon que les interactions entre groupes différents soient faibles (ex : réseau à routage hiérarchique)
    - faire l'estimation des paires OD conjointement au sein d'un même groupe mais indépendamment des autres groupes
  - avantages :
    - complexité numérique en $O(N_e^5)$ au lieu de $0(N_e^2)$ où N_e, nombre de noeuds de bordure
    - possibilité d'infrastructures d'estimation distribuées sur l'Internet
    - peu de dégradations par rapport à la méthode globale
  - limitations :
    - sur quel critère faire le clustering ?
      - information de Fischer
      - nombre de liens partagés en commun...
    - chaque sous problème est traité indépendamment des autres

Travaux à l'ENST Bretagne :

- Références :

  S.Vaton, A.Gravey, *Iterative Bayesian analysis of Network Traffic Data in the Case of Bursty Flows*, Internet Measurement Workshop, Marseille, nov. 2002.

- Généralisation de la méthode de *Metropolis within Gibbs* de C.Tebaldi au cas de mesures réitérées sur le réseau.

- PARTIE I :
  - les valeurs successives des flots OD sont indépendantes et identiquement distribuées selon une loi de Poisson
  - méthode Gibbsian EM pour estimer itérativement le débit moyen de chaque flot OD

- PARTIE II :
  - les flots OD suivent une dynamique markovienne
  - méthode d'estimation de type "turbo"

PARTIE I

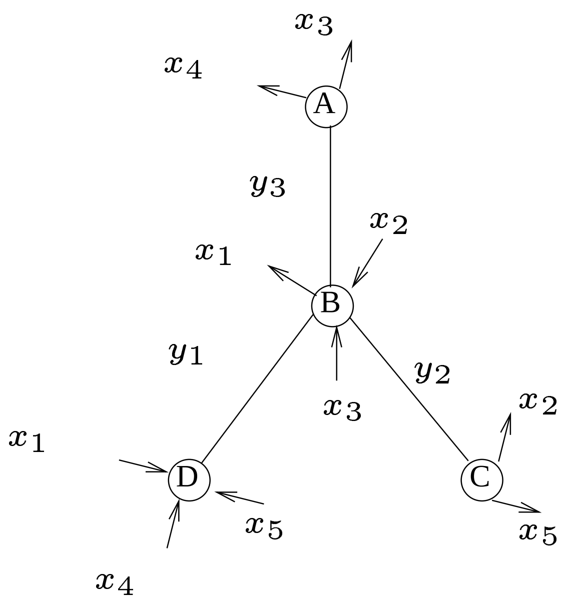
- Les valeurs successives prises par un même flot OD sont indépendantes et identiquement distribuées :

$$x_i^{(k)}, k = 1, K \sim \text{iid}\mathcal{P}(\lambda_i)$$

- méthode Tebaldi : produit une chaîne de Markov qui converge en distribution vers $p(\mathbf{x}^{(k)} \mid \mathbf{y}^{(k)} \; ; \lambda)$ en notant $\lambda = (\lambda_1, \cdots, \lambda_c)$
- Algorithme EM Gibbsien :
  - Étape E : calculer $\mathbb{E}(\mathbf{x}^{(k)} \mid \mathbf{y}^{(k)} \; ; \lambda^n)$ pour k=1,K (méthode MCMC à la Tebaldi)
  - Étape M : réactualiser l'estimateur de λ

$$\lambda^{n+1} = \frac{1}{K} \sum_{k=1}^{K} \mathbb{E}(\mathbf{x}^{(k)} \mid \mathbf{y}^{(k)} \; ; \lambda^n)$$

Exemple 1 :



5 paires Origine Destination, 3 liens

$y_1 \;: D \to B$	$x_1 : D \to B$	$y_1 = x_1 + x_4 + x_5$
$y_2 \;: B \to C$	$x_2 \;: B \to C$	$y_2 = x_2 + x_5$
$y_3 \;: B \to A$	$x_3 \;: B \to A$	$y_3 = x_3 + x_4$
	$x_4 \;: D \to B \to A$	
	$x_5 \;: D \to B \to C$	

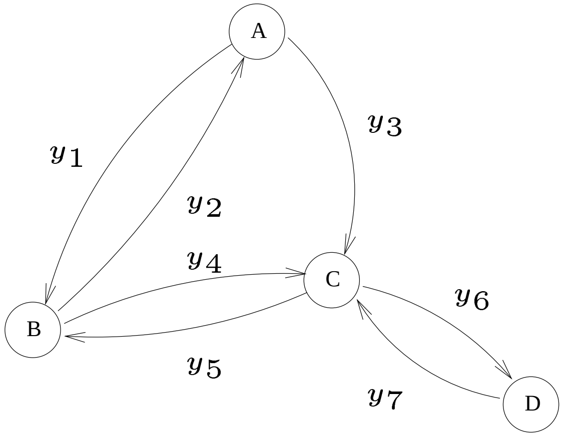
500 mesures réitérées sur le réseau ($K = 500$)

λ					
Vrai	25	30	12,5	6	32,5
Init.	21	21	21	21	21
it # 1	22	32	8	10	31
it # 2	22	32	8	10	31

$\mathrm{corr}(x_i, \hat{x}_i)$			
	it. # 0	it. # 1	it. # 2
x_1	0,68	0,72	0,72
x_2	0,78	0,81	0,81
x_3	0.77	0,79	0,79
x_4	0,46	0,59	0,59
x_5	0,78	0,83	0,83

- dans ce cas simple, pas de sensibilité au point d'initialisation
- convergence très rapide (1 itération suffit)
- amélioration modérée par rapport à la méthode non itérative de Tebaldi
- les flots dont la valeur de débit est faible par rapport aux autres flots sont souvent les moins bien estimés (ex. paire OD numéro 4) $\rightarrow$ ils ont tendance à être surestimés

Example 2 :



12 paires Origine Destination, 7 liens

$$
\begin{array}{lcl}
y_1 & : & A \to B \\
y_2 & : & B \to A \\
y_3 & : & A \to C \\
y_4 & : & B \to C \\
y_5 & : & C \to B \\
y_6 & : & C \to D \\
y_7 & : & D \to C
\end{array}
\qquad
\begin{array}{lcl}
x_1 & : & A \to B \\
x_2 & : & A \to C \\
x_3 & : & A \to C \to D \\
x_4 & : & B \to A \\
x_5 & : & B \to C \\
x_6 & : & B \to A \to C \to D \\
x_7 & : & C \to B \to A \\
x_8 & : & C \to B \\
x_9 & : & C \to D \\
x_{10} & : & D \to C \to B \to A \\
x_{11} & : & D \to C \to B \\
x_{12} & : & D \to C
\end{array}
$$

Matrice de routage :

$$
A = \begin{bmatrix}
\mathbf{1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \mathbf{1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1
\end{bmatrix}
$$

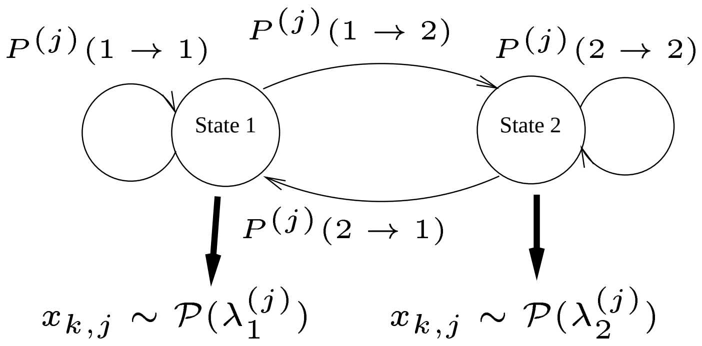
λ												
Vrai	22	25	10	30	42	27	35	35	25	50	65	40
Init.	30	30	15	30	40	30	30	30	30	40	40	40
it # 1	22	24	15	26	42	38	33	37	25	50	54	35
it # 2	22	24	15	26	42	38	33	37	25	50	54	35

corr(x_i, $\hat{x}_i$)			
	it. # 0	it. # 1	it. # 2
paire OD # 1	1,0	1,0	1,0
paire OD # 2	0,52	0,55	0,57
paire OD # 3	0,23	0,23	0,29
paire OD # 4	0,38	0,39	0,40
paire OD # 5	1,0	1,0	1,0
paire OD #6	0,50	0,46	0,46
paire OD #7	0,52	0,46	0,48
paire OD #8	0,55	0,55	0,54
paire OD #9	0,44	0,33	0,39
paire OD #10	0,73	0,75	0,76
paire OD #11	0,58	0,68	0,68
paire OD #12	0,68	0,67	0,68

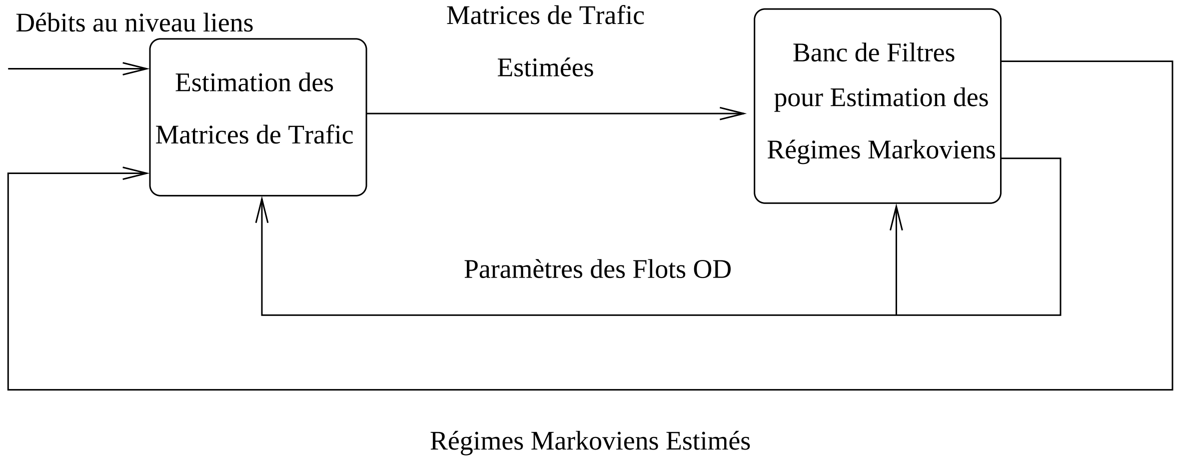
PARTIE II

Estimation de la matrice de trafic dans le cas de flots OD markoviens.

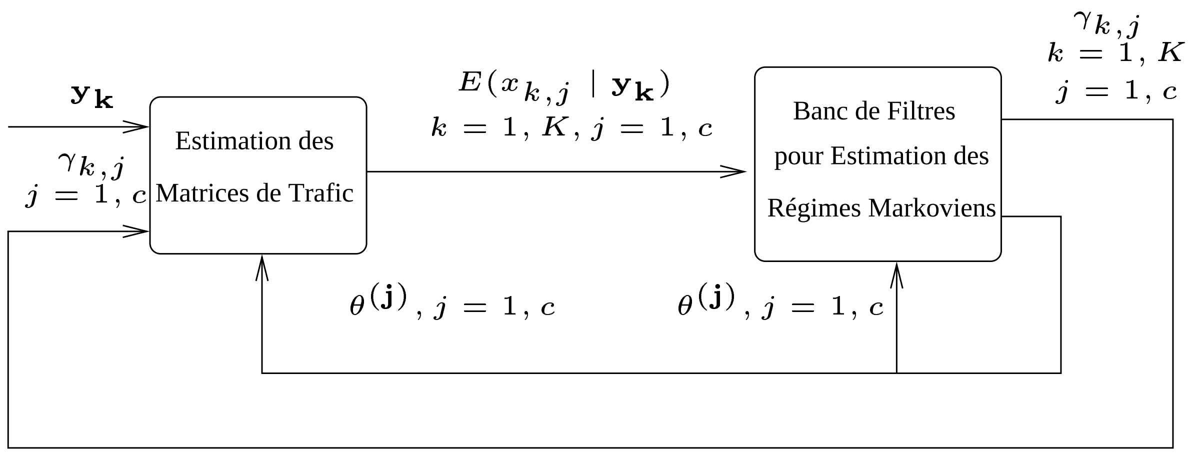
Exemple : diagramme de transition d'états du Switched Poisson Process à 2 états



Principe :



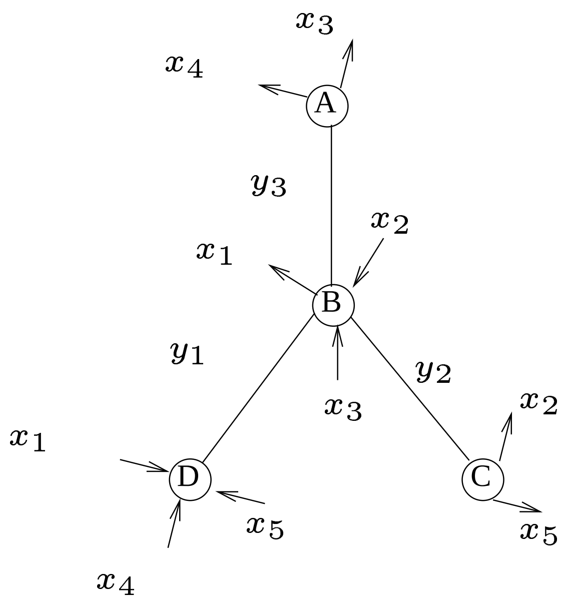
La boucle d'estimation.



On note $\gamma_k, j(s)$ la probabilité que le flot OD j soit dans l'état markovien s sur la fenêtre de mesure k. Cette quantité est obtenue par un algorithme avant arrière (*forward backward* algorithm, Baum et Welches).

On note θ_j le vecteur des paramètres statistiques du flot OD j (par exemple, dans le cas du Switched Poisson Process, θ_j représente les valeurs de λ et les temps de séjour moyen dans chaque état).

Exemple 1 : 3 liens, 5 paires OD. $K = 500$ mesures successives.



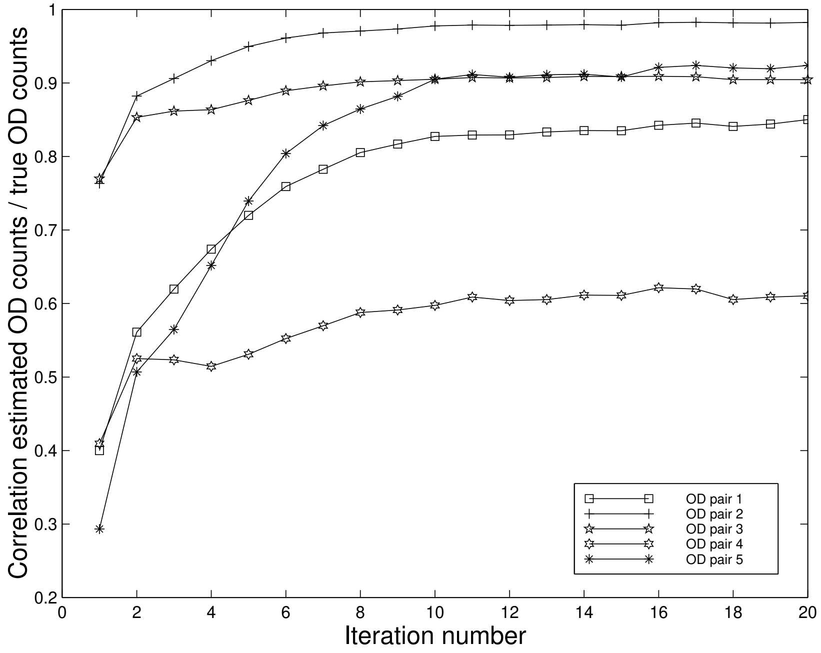
Avec une “bonne” initialisation des $\lambda_i^{(s)}$.

	Vrai		Init.		Estimé (20 itérations)	
	$\lambda^{(1)}$	$\lambda^{(2)}$	$\lambda^{(1)}$	$\lambda^{(2)}$	$\lambda^{(1)}$	$\lambda^{(2)}$
x_1	20	30	10	40	25.6	34.7
x_2	10	50	5	15	14.5	54.0
x_3	5	20	10	20	5.9	20.6
x_4	2	10	5	20	7.0	9.2
x_5	25	40	20	50	20.9	35.7

Intensités des flots OD dans les états “bas” ($\lambda^{(1)}$) et “haut” ($\lambda^{(2)}$).

	Vrai		Init.		Estimé (20 iterations)	
	TSM État 1	TSM État 2	TSM État 1	TSM État 2	TSM État 1	TSM État 2
x_1	50	50	100	100	62	52
x_2	50	50	100	100	30	48
x_3	50	50	100	100	22	128
x_4	50	50	100	100	400	∞
x_5	50	50	100	100	76	63

Temps de Séjour Moyen (TSM) dans l’état “bas” (état 1) et “haut” (état 2) (exprimé en nombre de périodes de mesure).



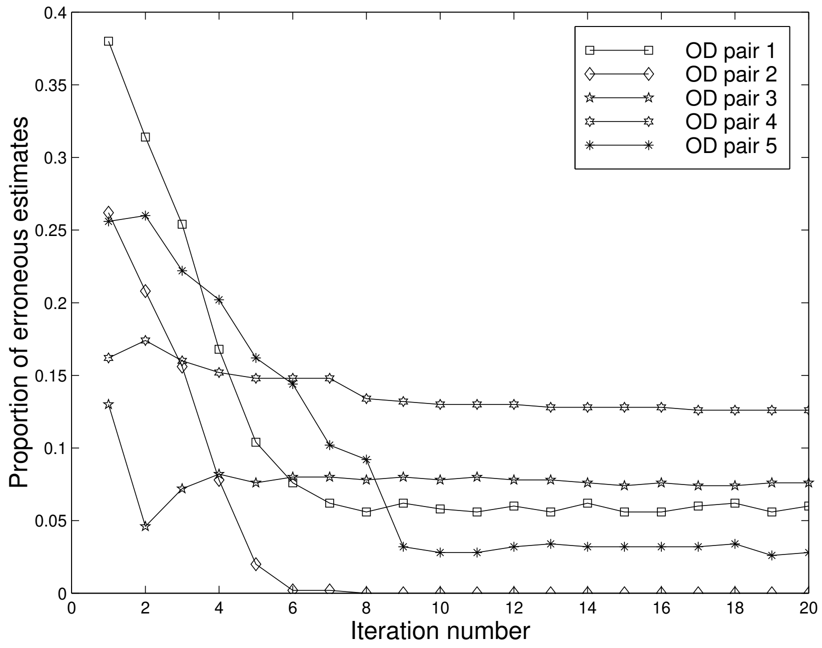


Figure 1: Corrélation entre débits OD réels et estimés.

Figure 2: Proportion des régimes markoviens qui sont mal estimés.

Avec une mauvaise initialisation : toutes les paires OD sont initialisées à la même valeur.

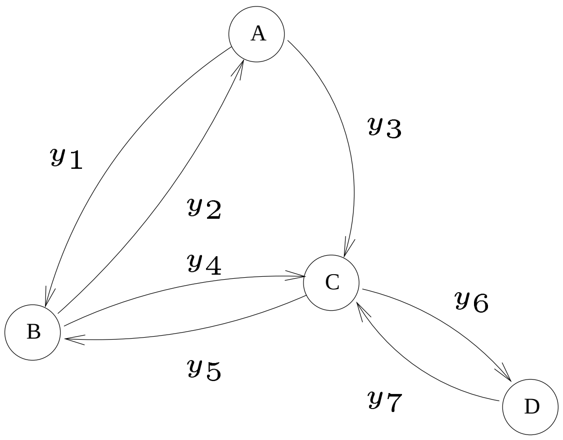
	Vrai		Init.		Estimé (20 itérations)	
	$\lambda^{(1)}$	$\lambda^{(2)}$	$\lambda^{(1)}$	$\lambda^{(2)}$	$\lambda^{(1)}$	$\lambda^{(2)}$
x_1	20	30	12.4	30	22.7	35.1
x_2	10	50	12.4	30	14.3	53.2
x_3	5	20	12.4	30	7.3	21.0
x_4	2	10	12.4	30	8.3	18.9
x_5	25	40	12.4	30	21.9	34.9

Intensités des flots OD.

	Vrai		Init.		Estimé (20 iterations)	
	TSM État 1	TSM État 2	TSM État 1	TSM État 2	TSM État 1	TSM État 2
x_1	50	50	100	100	30	47
x_2	50	50	100	100	33	38
x_3	50	50	100	100	22	28.5
x_4	50	50	100	100	∞	4
x_5	50	50	100	100	49	47

Temps de Séjour Moyen (TSM) dans l'état 1 et dans l'état 2
(exprimé en nombre de périodes de mesure).

Exemple 2 : 7 liens, 12 paires OD. $K = 500$ périodes de mesure.



$y_1 : A \to B$	$x_1 : A \to B$	$x_8 : C \to B$
$y_2 : B \to A$	$x_2 : A \to C$	$x_9 : C \to D$
$y_3 : A \to C$	$x_3 : A \to C \to D$	$x_{10} : D \to C \to B \to A$
$y_4 : B \to C$	$x_4 : B \to A$	$x_{11} : D \to C \to B$
$y_5 : C \to B$	$x_5 : B \to C$	$x_{12} : D \to C$
$y_6 : C \to D$	$x_6 : B \to A \to C \to D$	
$y_7 : D \to C$	$x_7 : C \to B \to A$	

Avec une “mauvaise” initialisation (toutes les paires OD initialisées à la même valeur).

	Vrai		Init.		Estimé (20 itérations)	
	$\lambda^{(1)}$	$\lambda^{(2)}$	$\lambda^{(1)}$	$\lambda^{(2)}$	$\lambda^{(1)}$	$\lambda^{(2)}$
x_1	15	30	20.6	48.0	15.1	30.1
x_2	10	40	20.6	48.0	10.6	40.5
x_3	2	20	20.6	48.0	20.0	36.5
x_4	20	40	20.6	48.0	22.1	41.9
x_5	15	80	20.6	48.0	15.0	79.5
x_6	5	50	20.6	48.0	26.4	66.0
x_7	30	40	20.6	48.0	24.7	37.1
x_8	10	60	20.6	48.0	19.6	59.6
x_9	20	30	20.6	48.0	18.8	41.7
x_{10}	40	60	20.6	48.0	42.2	60.5
x_{11}	50	80	20.6	48.0	26.2	51.8
x_{12}	30	50	20.6	48.0	16.9	33.7

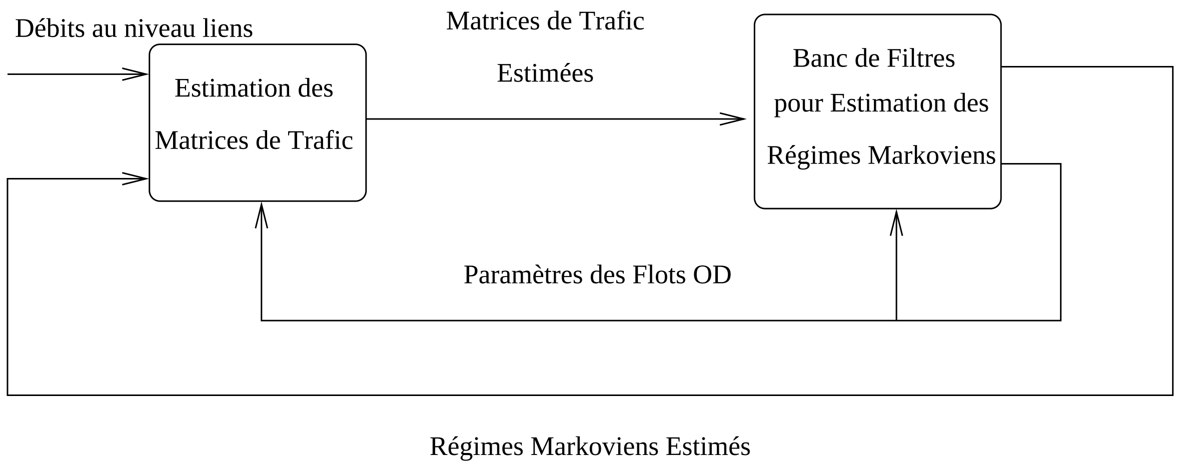
Intensités des flots OD.

	Vrai		Init.		Estimé (20 itérations)	
	TSM État 1	TSM État 2	TSM État 1	TSM État 2	TSM État 1	TSM État 2
x_1	50	50	100	100	56.8	18.6
x_2	50	50	100	100	21.8	21.8
x_3	50	50	100	100	10.8	40.3
x_4	50	50	100	100	9.43	15.1
x_5	50	50	100	100	26.0	26.7
x_6	50	50	100	100	14.0	18.1
x_7	50	50	100	100	32.8	35.7
x_8	50	50	100	100	51.8	37.7
x_9	50	50	100	100	144.9	19.0
x_{10}	50	50	100	100	28.1	36.5
x_{11}	50	50	100	100	10.2	23.6
x_{12}	50	50	100	100	59.2	65.8

Temps de Séjour Moyen (TSM) dans les États 1 et 2.

Conclusion

- estimation des matrices de trafic dans le cas de flots OD markoviens,
- idée fondamentale : "renvoyer" l'estimation des régimes markoviens sous la forme d'un a priori bayésien qui va aider la séparation des flots OD



La boucle d'estimation.

- on arrive à de meilleures performances avec le modèle markovien qu'avec le modèle indépendant

  $\hookrightarrow$ on est certainement intérêt à utiliser la dépendance temporelle de chaque flot OD pour améliorer l'estimation de la matrice de trafic

  $\hookrightarrow$ oui, mais quel modèle de flot ?

- Perspectives :
  - généralisation du principe “turbo” à d'autres modèles de flots OD
  - généralisation à même principe à d'autres estimateurs
  - validation sur données réelles