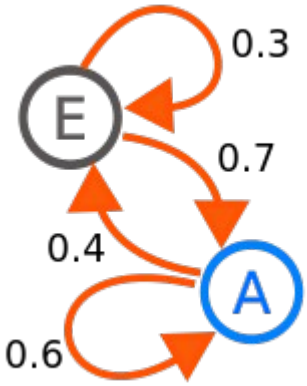
Cours : Performances des réseaux

I - Introduction, notion de critère de performance

Ère du big Data – Capacité de traitement / calculs
- grande capacité de transmission

Outils d'analyse de big Data.
Utilisé pour étudier système complexe : réseau routier, réseau d'interconnexion entre les gènes...
Basé sur une approche probabiliste des graphes. Utilise aussi les chaînes de Markov.



Exemple de chaîne de Markov

- *Exemple 1 :* Cas d'un E.N.T. (Espace Numérique de Travail)

Le serveur E.N.T. accède à des serveurs de stockage, d'applications pédagogique et réseau.

On observe de mauvaise performance, Est-ce due :
- à un serveur surchargé ?
- À un lien surchargé ?
- À un problème de synchronisation : tout le monde veux accéder à la ressources en même temps.

- *Exemple 2 :* cas d'un commutateur

N entrées et N sorties reliés les une aux autres

- Est-ce qu'on sert les paquets par nombre d'arrivée
- Est-ce qu'on sert les paquets les plus petits d'abord

…

- Exemple 3 : cas d'un nouveau protocole D2D (Device to Device)
- Quelle est la surcharge de ce protocole sur le réseau.

- Outils disponible :
- Simulation : « facile » mais lourd à développer. (Notamment la génération d'aléatoire difficile)
- Mesure : Donne les résultats du système en opération (ou en test) mais lourd à mettre

en œuvre.

- Analytique : résultats rapides mais peut devenir vite complexe, mais permet de connaître les grandes lois du trafic.

- Dans ce cours :	- Analyse opérationnelle	Une F.A. : ---buffer----serveur---
	- Les chaînes de Markov	
	- F.A. simples (Files d'Attente)	
	- Réseau de F.A.	

/!\ pour le CF : On a le droit à une antisèche A4 recto-verso.

Notion de critère de performance

- Comme en mécanique on définit un système: réseau, FA simple, Serveurs, ect.
- Dans ce système arrive des clients.
- Ce système à des paramètre modulable par l'admin : vitesse de traitement de serveur, débit d'un lien, taille de réseau, ect.
- On observe des performances mesuré par des vitesses de performances.

- TEMPS DE RÉPONSE (Moyen): R
C'est le temps qui s'écoule entre l'instant d'arrivée d'un client sur le syst & le moment où il la quitte.

- NOMBRE MOYEN DE CLIENTS DANS LE SYSTÈME :

- DÉBIT SORTANT DU SYSTÈME (*RQ* . : le temps entre deux départs du système est 1/R)

- TAUX D'OCCUPATION : - Probabilité pour qu'une ressource soit occupée.
- On essaye de maximiser le taux d'occupation d'une ressource chère (le gaspillage c'est mal !)

- LE TAUX DE PERTES :
- Proba qu'un client soit perdu
- proba qu'un client soit rejeté sachant qu'il arrive.

II – ANALYSE OPÉRATIONNELLE

1. Formule de Little opérationnelle

Soit un système ouvert. Les clients y arrivent par l'entrée, subissent un traitement puis sortent.
On ne fait aucune hypothèse sur « l'intérieur » du système. (e.g. sur l'ordre dans lequel les clients sont servis)

R : temps de réponse moyen
L : nombre moyen de clients dans le système
Λ : débit sortant

Little affirme : $L = \Lambda * R$

Supposons qu'on fait une mesure pendant une durée T.

$L = \frac{\Sigma k * Q(k)}{T}$ où Q(k) est le temps total pendant lequel il y a eu k clients dans le système.

Soit N(T) le nombre de clients au cours du temps :

$\Lambda(T) = \frac{N(T)}{T}$

RQ1 : L est en générale + facile à mesurer que R.
Mesurer le temps de réponse nécessite d'enregistrer toutes les dates d'arrivées &, lors d'une sortie, de mettre à jour une variable $X+=(t_{sortie} - t_{arrivé}) \rightarrow R = X / T$
Il est + facile de mettre à jour une variable à chaque changement de la longueur de la file d'attente.

RQ2 : On s'intéresse en performance à des moyennes en probabilité (e.g. E[L] qu'on note abusivement L)

$L(t) = \int_0^t \frac{l(u)}{t} du$ où l(u) est le nombre de clients à l'instant u du système.

De manière générale, surtout pour les chaînes de Markov, on regarde l'espérance comme limite de moyenne temporelle : $E[L(t)] = L = \lim_{t \to +\infty} L(t) = \int_0^t \frac{l(u)}{t} du$

C'est vrai quand les hypothèses du théorème ergodique sont vérifiées.

Application de la formule de Little aux réseaux fermés:
Supposons qu'on connaisse Λ, N (le nombre de terminaux) et Z (le temps de services d'un terminal)
Que vaut R le temps de réponse ?

$R + Z = \frac{N}{\Lambda} \rightarrow R = \frac{N}{\Lambda} - Z$

2. Relation de Chang-Lavenberg

Soit un réseau composés de files d'attentes. Les clients envoyés dans ce réseau peuvent engendrer plusieurs requêtes sur les stations, ce qu'on modélise par le fait qu'un client va passer ei fois en moyenne dans la station.

On suppose que les stations ne peuvent servir qu'un seul client à la fois. L'ordre de service n'est pas forcément 1er arrivé, 1er servi dans les files d'attentes.

On note :
L_i : nombre moyen de client dans la FA_i
R_i : temps de réponse moyen dans la FA_i
U_i : taux d'occupation dans la FA_i
Λ_i : débit d'arrivée sur la station $_i$
Λ : débit d'arrivée sur le réseau
S_i : temps de service de la FA_i

On a alors $U_i = \Lambda * e_i * S_i$

$U_i = \Lambda_i * S_i$

- Démo :

On a : $U_i = \frac{T - Q_i(0)}{T}$

$S_i = \frac{T - Q_i(0)}{N_i(T)}$ où $N_i(T)$ est le nombre de clients parties de la station i pendant T.

→ $$U_i = \frac{T - Q_i(0)}{N_i(T)} * \frac{N_i(T)}{T} = S_i * \Lambda_i = S_i * \Lambda * e_i$$

RQ1 : Si l'on suppose qu'un même client ne peut être servi par plusieurs F.A. à la fois :

$$L = \sum_i L_i$$

$$R = \sum_i e_i * R_i$$

Notion de charge en Eriang

La quantité $\Lambda_i * S_i = e_i * \Lambda * S_i$ est égale à U_i

Elle représente la demande de travail par unité de temps à la F.A. i

Si l'on note $\mu_i = \frac{i}{S_i}$ le taux de service

$\Lambda_i * S_i = \frac{\Lambda_i * e_i}{\mu_i}$ peut être vu comme la quantité de travail (ou charge) demandé à la station i rapportée à la capacité de traitement de la station i.

C'est une charge normalisée dont l'unité s'appelle l'Eriang (Er).

Exemple : un lien à 10 Gb/s chargé avec un trafic à 1 Gb/s supporte 0,1 Er

Étude de la saturation d'un système, notion de goulot d'étranglement :

Supposons qu'on ait un réseau comme ci-dessus. Le taux d'occupation de la station i est :

$U_i = \Lambda * e_i * S_i$

Il y en a (au moins) une qui est plus chargée que les autres, c'est ce qu'on appelle le goulot d'étranglement du support.

On ne peut évidemment pas dépasser $U_i = 1$. C'est le régime (limite) de saturation.

Dans la pratique il n'existe pas de transition nette entre le régime non saturé et le régime saturé. Un système chargé à 80 % est déjà bien souvent saturé.

RQ2 : Le débit de sortie de la file saturée n'est pas nécessairement μ_i dans la pratique.

3. Temps résiduel de service

Supposons qu'on ait des arrivées de clients séparées par des temps X' aléatoire.

Ces clients attendent une ressource qui vient régulièrement, séparée par des temps X aléatoire.

Exemple : Des passagers arrivent en gare (séparés par des temps X')

Des trains les prennent (séparés par des temps Y)

Combien de temps un client qui arrive attend il le passage de la ressource

(/!\ l'attente n'est pas due à un autre client traité /!\ )

- Formule de Pollaczek-Khintchine :

$$E[Y]=E[x]*\left(\frac{1+CCV_X}{2}\right) \quad \text{où} \quad CCV_X=\frac{VAR[X]}{E[X]^2}=\frac{E[X^2]-E[X]^2}{E[X]^2}$$

Chaîne de Markov à Temps Discret

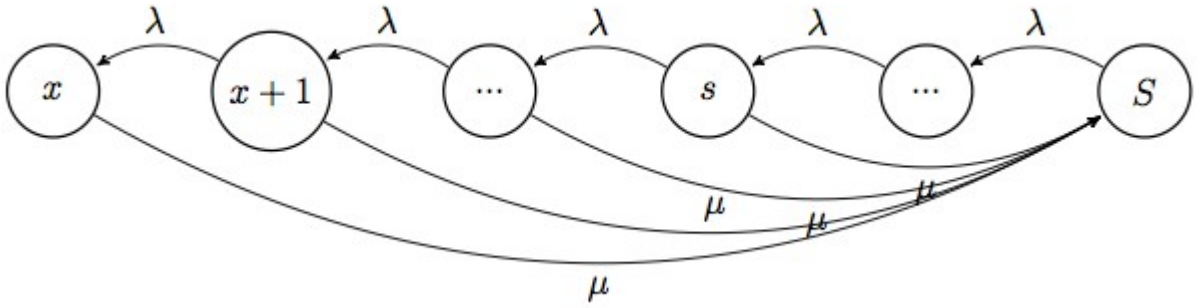
(processus sans mémoire) (CMTD)

On à un modèle général qui s'applique à beaucoup de cas (modèle classique)

→ On considère un système et aux cas possibles. (*Exemple :* un état = nombre de client dans la FA)

→ On veut un système discret d'état. Ceux-ci peuvent être compliqué mais ils doivent être dénombrable.

Exemple :



- **Un processus sans mémoire :** L'évolution du système suit une loi connue quand l'état présent est connu. Il n'y a pas besoin de connaître l'histoire passé.

On note X_n l'état du système à l'instant n.

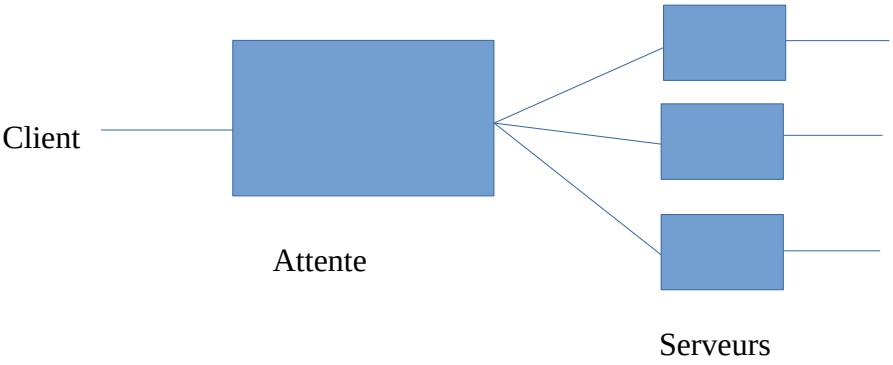
- **Probabilité que [X_n = « un état i »] = $\Pi^{(n)}(i)$**

On travail avec des **vecteurs de probabilité d'état** (Vecteur ligne à termes positifs dont la somme vaut 1) => **$\Pi^{(n)}$ = [$\Pi^{(n)}(0)$, …, $\Pi^{(n)}(i)$]**

Si $\Pi^{(n)}$ tend vers un vecteur, on dit que cette **chaîne de Markov est convergente**.

- **Un état** (ou ensemble d'état) **absorbant** est un état dont on ne sort plus un fois entré.
- **Un état transitoire** est un état non absorbant.

Définition et caractérisation des files d'attente



Notation de Kendall

A/B/N et éventuellement (capacité *[par défaut infini]* / Loi de priorité [par défaut FIFO] / taille du réseau si formé [par défaut infini])

A : Processus d'arrivé
→ loi du temps inter-arrivées successives
B : Loi de la durée des services
N : Nombre de serveurs

1. Pour la notation, on associe les lettre A,B,N à d'autre lettre pour caractériser la FA.
   - On peut associer A à M pour indiquer que c'est un processus de Markov (Ex. : loi de poisson sans mémoire) → $Pr[X \leqslant x] = 1 - e^{-\lambda x}$ où $E[X] = \frac{1}{\lambda}$ et λ est le débit ($CCV^2 = C^2[X] = 1$)
   - E : Erlang Σ de r variables exponentielles indépendantes

     ___X1___X2___ … Xr___ $X_E = \sum_{i=1}^{r} X_i$

     Voir d'écrire la loi d'Erlang

   - D : Déterministe
   - HK : Hyper exponentielle $C^2 > 1$

   → la classes de clients
   → arrivées groupées
   - G : général, on ne fait pas d'hypothèse
   - GI : général mais indépendantes

Principe de blocage : Si on précise une capacité, on ne prend pas les nouveaux clients (ils peuvent être perdus où alors on indique qu'on ne prend plus de nouveau clients jusqu'à avoir libéré de la place)

Exemple : M/M/I : arrivé Poisson, service exp, 1 serveur, FIFO, capacité ∞, file d'attente ouverte)

Il existe différente lois de priorité :
- FIFO
- LIFO
- Round Robin (Pas plus de service qu'un quantum)
  On interrompt le service au bout d'une durée q et on le remet dans la FA. Ainsi on traite en priorité les plus facile (rapide) à faire
- $Processor\ Sharing = \lim_{q \to 0} Round\ Robin$