Un algorithme de MBAC pour Cross-Protect

Dans ce chapitre, nous nous penchons sur la problématique bien connue du contrôle d'admission et, en particulier, la réalisation d'un contrôle d'admission basé sur des mesures (MBAC, *Measurement Based Admission Control*). Malgré une recherche abondante sur de nombreuses années, il est raisonnable de dire qu'aucun algorithme satisfaisant n'a été proposé pour l'admission des flots à débit variable, même dans le cas favorable d'un multiplexage sans buffer ("bufferless multiplexing"). Le sujet est clairement passé de mode mais le contrôle d'admission demeure un composant essentiel au contrôle de la qualité de service et de nombreuses questions restent posées. Nous considérons ici la conception de l'algorithme de MBAC dans le contexte particulier de Cross-Protect, qui est nécessairement basé sur les informations limitées sur le trafic offertes par l'ordonnanceur.

Sommaire

Contributions :

- Évaluation de l'applicabilité des algorithmes connus de MBAC au contexte de Cross-Protect ;
- Adaptation de l'algorithme de contrôle d'admission de Cross-Protect à un contexte plus réaliste de trafic et proposition de deux variantes ;
- Évaluation dans de nombreux scénarios de flots hétérogènes, gigués, ainsi qu'en régime non stationnaire de surcharge ;
- Étude de l'impact de l'intervalle d'échantillonnage des estimateurs.

L'implémentation des algorithmes développés et utilisés dans cette section ont été intégrés dans le module ns-2 Cross-Protect.

Publications et présentations :

Ce chapitre à fait l'objet de la publication suivante :

- Jordan Augé, Sara Oueslati, James Roberts, **Measurement-based admission control for flow-aware implicit service differentiation**, *23rd International Teletraffic Congress (ITC 2011), San Francisco (CA), Sep 6-8, 2011*.

En outre, un ensemble de résultats non présentés dans cette thèse ont fait l'objet des rapports techniques internes suivants :

- Sara Oueslati, Jordan Augé, Philippe Olivier, James Roberts, Raymond Cicutto, Jean-Louis Simon, **Premières analyses de la trace en surcharge IDF-La Réunion**, Rapport technique, Décembre 2005.
- Philippe Olivier, Jordan Augé, Sara Oueslati, **Étude préliminaire des caractéristiques du trafic IP sur un lien en surcharge**, Rapport Technique, Février 2006.

Ils décrivent la capture, l'analyse et le rejeu par simulation avec un contrôle d'admission d'une trace de trafic capturée sur un lien opérationnel en situation de surcharge, à la suite d'une panne de lien et d'un reroutage de son trafic.

6.1 Introduction

Le contrôle d'admission au sein de Cross-Protect, qui est un Contrôle d'Admission Basé sur des Mesures (MBAC, *Measurement-Based Admission Control*), joue le rôle essentiel de maintenir le *fair rate* suffisamment haut – afin de continuer à traiter les flots *streaming* en priorité et d'assurer un débit suffisant aux flots élastiques – et de conserver la charge de la file prioritaire en dessous d'un certain seuil. Il est soumis à la connaissance limitée du trafic dont nous disposons dans l'ordonnanceur. Il s'agit essentiellement du volume de paquets émis dans la file prioritaire lors des intervalles de temps successifs, ainsi qu'un estimateur du *fair rate* courant. Nous n'avons aucune connaissance *a priori* des caractéristiques du trafic et détectons seulement la terminaison des flots par l'absence de nouveaux paquets pendant un intervalle de temps donné (*timeout*).

Quand une proportion significative du trafic est élastique, et que les flots peuvent atteindre le *fair rate*, il est relativement facile de protéger la performance des flots en cours, simplement en rejetant les nouveaux flots lorsque le *fair rate* se retrouve en dessous d'un certain seuil (généralement de l'ordre de 1% de la capacité du lien [19] comme nous l'avons vu dans le Chapitre 3, Section 3.5.3, et rarement dépassé sauf en cas de surcharge importante). Nous considérons ici le cas plus problématique mais très probable où la grande majorité des flots possède un débit limité (par exemple à cause de leurs débits d'accès ADSL) et par conséquent se retrouve, dans des conditions de charge normale, traitée en priorité par l'ordonnanceur de Cross-Protect.

Bien que la connaissance des caractéristiques du trafic soit limitée, le contexte que nous considérons est très favorable à un multiplexage statistique efficace. Par hypothèse, le débit du lien est bien supérieur au débit crête maximum des flots devant recevoir un service prioritaire. Ce dernier est nécessairement inférieur au seuil sur le *fair rate* aux alentours de 1%, et même bien inférieur encore la plupart du temps (par exemple, en considérant des flots vidéo à 4Mb/s qui se partagent un lien de cœur de réseau OC196 à 10Gb/s). Il est ainsi possible de maintenir une forte utilisation des ressources tout en garantissant une performance excellente pour chaque flot.

Dans la suite de chapitre, nous discutons le choix d'un critère d'admission approprié pour préserver la qualité des flots *streaming* dont le débit crête est inférieur à un seuil dénoté $p < FR$. Nous proposons un algorithme simple et illustrons ses performances aux travers de nombreuses simulations, en régime de surcharge et dans des conditions de *flash crowds*.

6.2 État de l'art des approches de MBAC

De nombreux algorithmes de MBAC ont été présentés et étudiés au fil des années. Cependant, il n'en existe qu'un petit nombre qui se satisfait des hypothèses minimales sur le trafic qui conviennent dans notre contexte d'étude. Nous limitons cette section à ces contributions uniquement.

6.2.1 Measured sum

Jamin et al. [76] proposent un algorithme simple de MBAC appelé *Measured Sum* (MS). Un nouveau flot est admis si la somme de son débit nominal r et du débit estimé de l'agrégat des flots en cours $\widehat{v}$ est inférieure à un seuil d'utilisation u de la capacité du lien C, comme illustré par l'équation 6.1.

$$\widehat{v} + r \leq uC \tag{6.1}$$

La bande passante résiduelle constitue une marge de sécurité qui permet d'absorber les fluctuations du trafic. Le débit agrégé des flots est mesuré à l'aide d'un estimateur sur une fenêtre glissante de T slots de durée S ; Casetti *et al.* [43] proposent une version de l'algorithme basée sur un ajustement adaptatif de la longueur de cette fenêtre. $\widehat{v}$ est mis à jour à la fin de chaque slot, ainsi qu'à l'arrivée d'un nouveau flot, où est il augmenté du débit crête du flot :

$$\widehat{v} = \begin{cases} MAX(\widehat{v}^S), & \text{sur une fenêtre de T slots} \\ \widehat{v}^S, & \text{si } \widehat{v}^S > \widehat{v}, \text{ où } \widehat{v}^S \text{ est le} \\ & \text{débit mesuré sur une période S} \\ \widehat{v} + r & \text{à l'admission d'un nouveau flot} \end{cases}$$

Cet algorithme présente l'avantage d'être simple malgré la sensibilité aux paramètres S et T, et il ne requiert que la connaissance du débit crête des flots. La performance réalisée (taux de pertes) est cependant peu prévisible (au mieux la performance du pire cas, qui suppose un débit crête maximal afin de calculer le taux d'utilisation maximal cible).

6.2.2 Borne de Hoeffding

Le contrôle d'admission introduit par S. Floyd [55] calcule la bande passante équivalente d'un ensemble de flots (une estimation de la bande passante agrégée) en utilisant la borne de Hoeffding (un résultat de la théorie des probabilités qui donne une borne supérieure à la probabilité que la somme de variables aléatoires dépasse un certain seuil).

Cette bande passante équivalente dépend du choix d'une probabilité de perte cible et s'exprime ainsi :

$$\hat{C_H}(\hat{\mu}, \{p_i\}_{1 \leq i \leq n}, \epsilon) = \hat{\mu} + \sqrt{\frac{\ln 1/\epsilon \sum_{i=1}^{n} (p_i)^2}{2}}$$

$\hat{\mu}$ est un lissage exponentiel du débit d'entrée agrégé, ϵ est la probabilité de perte cible, p_i le débit crête du flot i, et n le nombre de flots en cours.

Un nouveau flot est admis si la somme de son débit crête et de la mesure de la bande passante équivalente est inférieure à la capacité du lien :

$$\hat{C_H}(\hat{\mu}, \{p_i\}_{1 \leq i \leq n}, \epsilon) + p_{n+1} \leq C$$

Comme pour l'algorithme MS, l'estimation du débit entrant agrégé $\hat{\mu}$ est augmentée à chaque addition du débit crête du nouveau flot.

L'algorithme a l'avantage d'intégrer explicitement la probabilité de perte cible dans les conditions d'admission. La borne de Hoeffding n'est cependant pas une borne forte, ce qui conduit à une performance conservatrice de l'algorithme, c'est-à-dire une utilisation faible. La méthode suppose également la connaissance du débit crête individuel de chaque flot, et encore plus difficile, le nombre de flots en cours. De plus les calculs de bande passante équivalente sont basés sur des flots à débit constant, ce qui est loin d'être le cas en réalité.

6.2.3 Calcul d'enveloppes de trafic

La méthode de Qiu et Knightly [127] utilise des mesures de l'enveloppe maximale d'un agrégat de trafic. La condition d'admission est calculée à l'aide de la moyenne et de la variance de ces enveloppes, ainsi que d'une probabilité de perte cible ; un nouveau flot de débit crête p est admis si :

$$\overline{R}_k + p + \alpha\sigma_k \leq C, \forall k = 1, 2, .., T$$

où $\overline{R}_k$ est l'estimation du débit crête entrant agrégé, mesuré pour différentes échelles de temps – k variant de 1 à T slots dans la fenêtre courante (de T slots) –, et σ_k est la variance de ces mêmes mesures sur les M dernières fenêtres. $\alpha = Q^{-1}(p_q)$, avec p_q la probabilité de perte cible et $Q(.)$ la fonction de distribution complémentaire d'une variable aléatoire Gaussienne $N(0,1)$.

Un niveau de confiance est également calculé afin de pallier aux incertitudes des mesures. L'enveloppe est utilisée afin de capturer la variabilité du trafic à différentes échelles de temps. Toutefois, ils utilisent un modèle avec buffer, qui le rend dépendant de caractéristiques détaillées du trafic. Le processus de mesure est de plus relativement complexe à cause des nombreuses échelles de temps nécessaires.

6.2.4 Approche basée sur la théorie de la décision

Gibbens et al. [62] proposent une approche basée sur la théorie de la décision, où un nouveau flot est admis si la charge agrégée courante est inférieure à un seuil approprié. Parmi les méthodes proposées, la plus simple ne prend en compte que le débit crête d'un flot.

Cela correspond exactement à nos besoins et les résultats confirment qu'un multiplexage efficace est possible dans un tel cadre d'étude. Malheureusement, le calcul du seuil nécessite des hypothèses sur le niveau de charge offerte, ainsi que sur la *burstiness* des flots (le degré de rafales de paquets).

6.2.5 Théorie de la bande passante équivalente

Gibbens et Kelly [61] présentent une famille d'algorithmes basées sur des tangentes à une courbe de bande passante équivalente, calculée à partir de la borne de Chernoff (qui améliore la borne de Hoeffding).

Ici, différents choix de tangentes impliquent la connaissance de différentes caractéristiques du trafic. Les flots sont supposés appartenir à un nombre donné de classes. L'approche la plus simple nécessite la connaissance de la charge globale instantanée (comme dans [62]) ainsi que le nombre de flots de chaque classe accompagné de leur débit crête. Cette méthode s'apparente à l'utilisation de la borne de Hoeffding proposée par Floyd [55].

La méthode de la tangente au débit crête admet un nouveau flot si :

$$np(1-e^{-sp})+e^{-sp}\widehat{v}\leq C$$

où n est le nombre de flots admis, p le débit crête, s le paramètre spatial de la borne de Chernoff, $\widehat{v}$ l'estimation de la charge globale et C la capacité du lien.

La complexité de ce genre d'algorithmes (basés sur des bornes de Chernoff) est trop importante pour qu'ils puissent être considérés pour la prise de décisions d'admissions en temps réel. De plus, les alternatives identifiées dans [61] requièrent toutes des informations supplémentaires qui ne sont pas disponibles dans Cross-Protect.

6.2.6 Stratégies de *back off*

Les méthodes précédentes, proposées par Gibbens *et al.* [62], Gibbens et Kelly [61] et Floyd [55] proposent toutes d'utiliser une stratégie de *back off* : lorsqu'un flot est bloqué, aucun autre flot n'est accepté tant que l'un des flots en cours ne termine pas. Cela évite le problème d'un grand nombre d'admissions dans des conditions de forte charge, résultant d'une seule mesure faible. Malheureusement un tel procédé n'est pas possible avec Cross-Protect puisque les terminaisons de flots ne sont pas explicitement connues.

6.2.7 Décomposition selon différentes échelles de temps

La notion d'échelle de temps critique

L'algorithme proposé par Grossglauser et Tse [65, 66] se base sur l'idée de décomposer les variations du débit agrégé selon deux échelles de temps : des fluctuations à court et à long termes. La durée critique séparant les deux échelles de temps est :

$$\tilde{T}_h := T_h/\sqrt{n}$$

où T_h est la durée moyenne des flots, n est l'estimation du nombre maximal de flots en cours et qui peut être écrit : $n=\sqrt{\frac{C}{\mu}}$, où μ est le débit moyen des flots. D'après les auteurs, les fluctuations du débit agrégé qui se produisent à une échelle de temps plus grande que $\tilde{T}_h$ sont automatiquement compensées par les arrivées et les départs de flots. Par conséquent, il suffit de réserver de la bande passante uniquement pour absorber les fluctuations à court terme, qui sont dues aux variations de débit intrinsèques aux flots en cours.

Une condition d'admission est déduite afin de satisfaire une probabilité de perte cible, sous les hypothèses du multiplexage sans buffer.

Leur première proposition suppose que les caractéristiques individuelles des flots, leur débit moyen et leur variance, sont mesurées. Un nouveau flot est admis si la condition suivante est satisfaite :

$$C-(N_t+1)\widehat{\mu}_t>\alpha_q\widehat{\sigma}_t^H\sqrt{N_t+1} \tag{6.2}$$

où C est la capacité du lien, N_t le nombre de flots dans le système au temps t, $\widehat{\mu}_t$ le débit moyen mesuré des flots, $\alpha_q=Q^{-1}(\epsilon)$, où ϵ est la probabilité de perte cible, $Q(.)$ la cdf complémentaire d'une variable aléatoire Gaussienne $N(0,1)$, et $\widehat{\sigma}_t^H$ l'écart type estimé du débit par flot.

Si S_t est le débit entrant agrégé au temps t, alors $\hat{\mu}_t$ est donné par :

$$\hat{\mu}_t=\int_0^\infty \frac{S_{t-\tau}}{N_{t-\tau}}g_\tau d\tau$$

où g_t est un filtre passe-bas, de fréquence de coupure $1/\tilde{T}_h$; et $\widehat{\sigma}_t^H$, l'estimation de la variance des hautes fréquences, est donnée par :

$$\hat{\sigma}_t^H=\left[\int_0^\infty\left[\frac{S_{t-\tau}^H}{N_{t-\tau}}-\int_0^\infty\frac{S_{t-u}^H}{N_{t-u}}h_u du\right]^2 h_\tau d\tau\right]^{1/2}$$

Nous ne pouvons pas utiliser cette méthode notamment parce que le nombre de flots en cours N_t n'est pas connu dans Cross-Protect.

Le deuxième article fournit une méthode qui est plus proche de nos hypothèses, lorsque N_t n'est pas connu, puisqu'elle utilise des mesures du débit agrégé en entrée, ainsi que le débit crête des flots admis. Leur méthode se base sur une approximation Gaussienne du débit agrégé. Un nouveau flot est admis si :

$$C - A_t^\lambda - p > \alpha_q \widehat{\sigma}_t^{AH} \tag{6.3}$$

où A_t^λ est la mesure agrégée du débit, p est le débit crête des flots et $\widehat{\sigma}_t^{AH}$ l'écart type du débit agrégé.

Pour obtenir cette formule , les variables A_L^t et A_H^t sont définies et correspondent à l'application de filtrages, respectivement passe-bas et passe-haut, aux fluctuations du débit entrant agrégé S_t :

$$\begin{aligned} A_t^L &= \int_0^\infty S_{t-\tau} g_\tau d\tau \\ A_t^H &= S_t - A_t^L \end{aligned}$$

où g_t est un filtre passe-bas.

Une autre variable $\hat{\sigma}_t^{AH}$ est introduite et correspond à l'estimation de la variance de la composante haute agrégée A_t^H :

$$\hat{\sigma}_t^{AH} = \left[\int_0^\infty \left[A_{t-\tau}^H - \int_0^\infty A_{t-u}^H h_u du \right]^2 h_\tau d\tau \right]^{1/2}$$

où h_t est également un filtre passe-bas.

L'estimateur passe-bas corrigé A_t^λ est dérivé ainsi :

$$A_t^\lambda = A_t^L + \lambda_t * g_t$$

le facteur de correction λ_t étant :

$$\lambda_t = \sum_i r\delta(t - t_i)$$

où t_i est la date d'admission du flot i, r son débit crête et δ la distribution de Dirac. Autrement dit, lorsqu'un flot est admis, son débit crête est ajouté à A_t^λ. Cela permet d'éviter des situations de surcharges momentanées à forte charge (leur hypothèse de trafic) dues aux admissions suivant un estimateur A_t^λ faible pendant une période t. Les instants d'arrivées de flots peuvent n'être connus que partiellement dans Cross-Protect puisque le contrôle d'admission est réalisé de manière distribuée sur plusieurs *line cards*.

Si l'on suppose que le débit agrégé en entrée ne change qu'en des instants discrets, alors A_t^λ peut être exprimé ainsi :

$$A_{t_i}^\lambda = \phi_i A_{t_{i-1}}^\lambda + (1 - \phi_i) S_{t_{i-1}} + r.\mathbf{1}_{\{\text{admission de flot à } t_i\}}$$

où t_i est l'instant où la bande passante agrégée S_t change ou bien qu'un nouveau flot est admis et

$$\phi_i = \exp\left(\frac{t_{i-1} - t_i}{\tilde{T}_h} \right)$$

.

La méthode proposée par Grossglauser et Tse [66, 65] est la plus proche de nos hypothèses sur le trafic. Elle possède notamment l'avantage, par rapport par exemple à [66], de ne réserver de la bande passante que pour les fluctuations du trafic à court terme, puisque celles à long terme peuvent être compensées par les arrivées et les départs de flots. Elle permet ainsi de parvenir à une meilleure utilisation du lien à condition de savoir déterminer l'échelle de temps critique $\tilde{T}_h$. Son calcul nécessite la connaissance de la durée moyenne ainsi que du débit moyen des flots, information qui n'est généralement pas disponible. Toutefois il est possible de contourner cet obstacle en mesurant par exemple $\tilde{T}_h$ en fonction des données passées. Nous pouvons alors profiter des nombreux avantages de cette méthode.

Knightly *et al.* utilisent une estimation classique non-biaisée, basée sur les données mesurées sur les M dernières fenêtres temporelles afin de prédire le trafic, tandis que Grossglauser et Tse utilisent un filtrage passe-bas à la fréquence de coupure $\tilde{T}_h^{-1}$ pour estimer $\hat{\mu}$, et à $T_s = k\tilde{T}_h^{-1}$ pour $\hat{\sigma}$. Cela permet une plus forte réactivité aux changement dans le profil de trafic et apparaît un choix plus raisonnable dans la mesure où les données récentes ont plus d'impact sur l'estimation du débit entrant que les plus vieilles.

6.2.8 Discussion

Malgré de grandes différences entre les algorithmes de MBAC, il s'avèrent qu'ils semblent tous conduire au même compromis entre le taux d'utilisation du lien et la performance perçue au niveau flot. Ceci est illustré dans [38] et [21] où la performance est mesurée en termes de taux de pertes de paquets. Les algorithmes diffèrent au niveau de la prévisibilité de leur performance : le choix de la valeur des paramètres pour la condition d'admission permettant d'obtenir un taux de performance cible. En fait Breslau et al. [38] montrent que tous les algorithmes qu'ils ont testés sont très peu efficaces à prédire la performance qui dépend de manière significative de caractéristiques du trafic qui ne sont pas explicitement prises en compte. Des raisons pour lesquelles il est extrêmement difficile de contrôler des indicateurs de performances tels que le taux de pertes de paquets au moyen d'algorithmes de MBAC sont évoqués par Bean [13]. Dans nos travaux, nous espérons ainsi aboutir à des algorithmes raisonnablement efficaces et c'est la limite de notre ambition.

6.3 Implémentation et évaluation de l'algorithme de Grossglauser et Tse

Une version discrétisée de l'algorithme de Grossglauser et Tse a été réalisée dans le simulateur `ns-2`. Nous la notons (`GT`). Les résultats obtenus avec cet algorithme nous serviront de référence lors de l'évaluation réalisée dans la section 6.5.

A la fin du (n+1)-ième intervalle, les nouvelles valeurs de la moyenne et de la déviation standard sont calculées ainsi :

$$\begin{aligned} A^L_{n+1} &= \alpha A^L_n + (1-\alpha) S_{n+1} \\ A^H_{n+1} &= S_{n+1} - A^L_{n+1} \\ D_{n+1} &= \beta D_n + (1-\beta)(A^H_{n+1})^2 \\ E_{n+1} &= \beta E_n + (1-\beta) A^H_{n+1} \\ \sigma_{n+1} &= (D_{n+1} - E^2_{n+1})^{1/2} \\ A^\lambda_{n+1} &= A^L_{n+1} \end{aligned}$$

A tout moment dans le (n+2)-ième intervalle, un flot peut être admis si :

$$A^\lambda_{n+1} + r < c - \alpha_q \sigma_{n+1}$$

Après une admission,

$$A^\lambda_{n+1} += r$$

Les paramètres de lissage des filtres passe-haut et passe-bas sont donnés par :

$$\begin{aligned} \alpha &= \exp\left(-\frac{\tau}{\tilde{T}_h}\right) \\ \beta &= \exp\left(-\frac{\tau}{T_s}\right) \end{aligned}$$

où T_s est environ 5 à 10 fois plus long que $\tilde{T}_h$.

Il convient de noter que λ_n est calculé à la fin du n-ième intervalle et est augmenté du débit crête des flots dans le $(n+1)$-ième comme suit :

$$A^\lambda_n = A^L_n + \lambda_n$$

La condition d'admission reste inchangée après l'admission d'un flot dans le $(n+1)$-ième :

$$A^\lambda_n += r$$

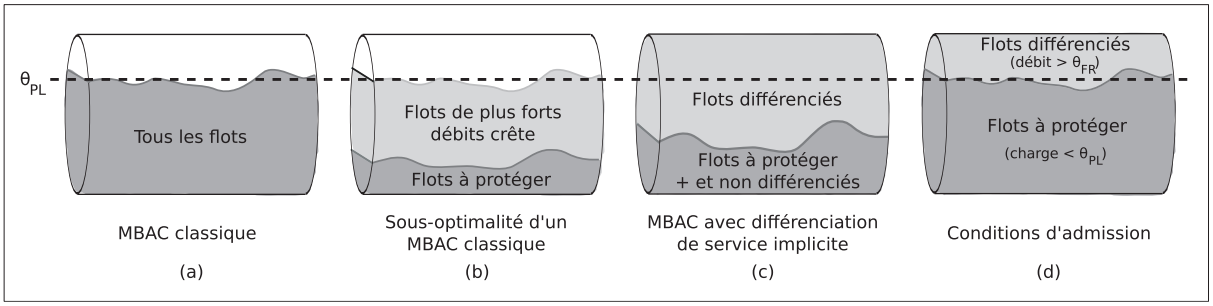


FIGURE 6.1 – Illustration de l'algorithme de MBAC avec divers régimes d'utilisation

6.4 Un algorithme de MBAC pour Cross-Protect

6.4.1 Critères d'admission

L'admission des flots dans Cross-Protect est déterminée par la valeur de deux mesures de congestion, le *fair rate* FR et le *priority load* PL [91]. Ces valeurs sont calculées par lissage exponentiel des valeurs mesurées sur des intervalles successifs de durée τ, et l'admissibilité dans l'intervalle t est déterminée par les valeurs calculées à $t-1$.

FR correspond au débit qu'un flot aurait en sortie s'il avait constamment des paquets à envoyer, et est estimé à partir des données disponibles dans l'ordonnanceur. Il est moyenné sur un intervalle en rapport avec l'échelle de temps des arrivées et des départs de flots. PL est la charge moyenne du trafic qui arrive dans la file prioritaire pendant chaque intervalle de temps. Les paquets envoyés dans la file prioritaire sont ceux qui n'appartiennent à aucun flot goulotté.

Quand une grande partie du trafic provient de flots élastiques qui ne sont nulle part limités en débit, le critère d'admission le plus significatif est FR. Le *priority load* reste bas de telle sorte que les délais et taux de perte des flots *streaming* de débit crête globalement inférieur au FR sont négligeables [91, 92]. Le contrôle d'admission est plutôt simple dans ce cas puisque les flots élastiques sont naturellement tolérants à une imprécision sur l'estimation du *fair rate*. La charge des flots *streaming* dans l'architecture Cross-Protect est ainsi initialement limitée à un seuil voisin de 70 ou 80%.

En pratique cependant, la grande majorité des flots élastiques possède un débit crête limité et, lorsqu'il est inférieur au seuil sur le *fair rate*, ceux-ci sont traités dans la file prioritaire conjointement aux flots *streaming*. Il s'agit par exemple de scénarios ADSL où les flots sont limités par leur débit d'accès. FR n'est alors plus critique tandis que le *priority load* mesuré PL inclue notamment des flots de débit supérieur à p que nous ne souhaitons pas protéger.

La difficulté pour le contrôle d'admission consiste à pouvoir différencier les flots en fonction de leur débit crête, sans avoir à les mesurer explicitement. Nous supposons que les flots de débit crête inférieur ou égal à p subiront des pertes et des délais négligeables si le trafic envoyé dans la file prioritaire de Cross-Protect reste inférieur à la capacité C du lien avec une probabilité d'au moins $1-\epsilon$. La valeur ϵ peut-être calibrée afin d'assurer des pertes et délais suffisamment bas.

6.4.2 Moyenne et variance du *priority load*

Notons $b(t)$ la mesure en bit/s de la charge envoyée dans la file prioritaire dans le slot t. Le *priority load* $B(t)$ est la moyenne glissante :

$$B_t = (1-\beta) \times B_{t-1} + \beta \times b_t. \tag{6.4}$$

où le paramètre $\beta = 1 - \tau/\tilde{T}_h$ et $\tilde{T}_h$ est l'échelle de temps critique définie dans [66].

Nous suivons l'approche des auteurs et proposons d'estimer la variance comme suit :

$$\widehat{\sigma}_t^2 = D_t - E_t^2 \text{ où} \tag{6.5}$$
$$D_t = (1-\gamma)D_{t-1} + \gamma(b_t - B_t)^2, \tag{6.6}$$
$$E_t = (1-\gamma)E_{t-1} + \gamma(b_t - B_t). \tag{6.7}$$

En suivant [66], le paramètre de lissage γ est choisi égal à $1-(1-\beta)/10$, et des mesures de $\tilde{T}_h$ peuvent être produites périodiquement (déduites lors des *timeouts*). En fait, l'estimation de cette dernière valeur reste problématique puisque la distribution de la durée des flots n'est pas exponentielle comme considérée dans [66] mais à queue lourde. Heureusement, il apparait de nos résultats non

présentés ici, ainsi que dans des travaux précédents, que le choix des paramètres β et γ n'est pas extrêmement critique.

L'utilisation de la condition d'admission (6.3) sur ces estimateurs est trop conservatrice pour les flots de débit crête supérieur à p. Nous supposons que de tels flots sont capables d'ajuster leur débit au *fair rate* en cas de surcharge (ou le devraient), et ne nécessitent pas d'être protégés. Par exemple, dans la figure 6.1b), les mesures du *fair rate* et du *priority load* sont les mêmes que dans la figure 6.1a), mais nous pourrions accepter plus de flots sans violer nos contraintes de délai pour les flots protégés. L'ajustement du seuil d'admission en tenant compte de la variance de l'ensemble des flots chercherait à maintenir la charge en dessous du seuil θ_{PL} et empêcherait le système d'entrer dans l'état favorable représenté dans la figure 6.1c). Dans cet état, les flots de débit supérieur à p deviennent *backlogged* et ne contribuent plus au *priority load*. Le dessin de la figure 6.1d) présente une situation idéale ou le contrôle d'admission différencie parfaitement les flots de débit crête inférieur et supérieur à p, et la composition du trafic est telle qu'elle permet au lien d'être saturé.

Nous voyons ici en quoi le contrôle d'admission de Cross-Protect diffère des algorithmes classiques qui opèrent en régime transparent. Ici, la saturation du lien est possible et même recommandée afin que les flots que nous ne souhaitons pas protéger soient différenciés.

6.4.3 Approximation basée sur une hypothèse Poissonnienne

Supposons que le nombre de flots inélastiques de débit crête inférieur ou égal à p en cours dans un intervalle donné a une distribution de Poisson. Ce serait le cas en l'absence de blocage, et dans le contexte du modèle de sessions Poisson (voir Chapitre 3). Si les paquets sont de taille constante maximale L, le nombre de paquets qui arrivent dans un intervalle plus petit que L/p possède également une distribution de Poisson [1]. Cela suggère qu'il est possible d'estimer la variance du débit dans un intervalle à partir de sa moyenne, ce qui permet de proposer le MBAC très simple qui suit, dénoté Poisson :

- Nous choisissons un intervalle de durée $\tau = L/p$.
- Étant donné la charge prioritaire mesurée B_t bits/s, $m_t = B_t\tau/L$ est une estimation du nombre de paquets, et nous déduisons l'estimation de la variance $\widehat{\sigma}_t^2 = m_t L^2/\tau^2 = B_t p$.
- Cette estimation peut alors être appliquée à la condition d'admission 6.3.

Performance du multiplexage statistique

En ignorant le blocage des flots et le fait que certains flots émettent des paquets de taille inférieure à L, l'approximation Poisson conduit à des décisions d'admission conservatrices. Cependant, p représentant un faible pourcentage de la capacité C, l'approximation permet une forte utilisation des ressources du lien. Le tableau 6.1 donne le seuil sur la charge du lien correspondant à des valeurs particulières de C/p et ϵ. Le contrôle d'admission serait appliqué dans l'intervalle t lorsque B_{t-1} dépasse ce seuil.

C/p	100	100	1000	1000
ϵ	0.001	0.01	0.001	0.01
α_q	3.09	2.33	3.09	2.33
Seuil	0.73	0.79	0.91	0.93

TABLE 6.1 – Seuils d'admission

Notons que si les flots de débit crête supérieur à p sont inclus dans l'estimation de la charge prioritaire B_t, la variance estimée par 6.5 sera plus grande que l'estimation Poisson $B_t p$. Le MBAC dérivé de [66] sera trop conservateur, préservant des états tels qu'en figure 6.1b). Nous espérons l'estimateur Poisson mieux capable de permettre des transitions vers des états tels qu'en figure 6.1c et d).

Le seul cas problématique pour la différenciation des flots est lorsque l'on a un mélange de flots qui ont en partie un débit inférieur à p et en partie supérieur. Toutefois, la contribution de ces flots en raison du carré de leur débit crête assurera une différenciation dans la majorité des cas.

Supposons que le débit crête des flots est $p = 100$kb/s, sur un lien de capacité $C = 10$Mb/s. Nous normalisons ces valeurs afin d'obtenir une capacité unitaire ($p = 10^{-2}$). Afin d'assurer une probabilité de perte $\epsilon = 10^{-2}$, nous avons vu que la charge du lien devait être limitée à une valeur de $\theta = 0.79$ telle que : $\theta + \alpha_q\sigma = 1$ avec $\sigma = \sqrt{\theta p}$.

1. Un flot de débit r contribue de manière indépendante 1 paquet avec la probabilité r/p.

Supposons maintenant qu'avec les mêmes paramètres, à la même charge, le trafic soit un mélange de 2 classes de flots, respectivement $P_1 = 50$ et $P_2 = 200$kb/s. Chacune contribue à la moitié de la charge générée.

Le calcul de la variance du trafic donne alors : $V = \frac{\theta}{2P_1} P_1^2 + \frac{\theta}{2P_2} P_2^2$, soit $\sigma' > \sigma$ et ainsi une probabilité de débordement supérieure qui se traduira par une différenciation des classes de flots avec une forte probabilité.

Intégration de la variance

Si le trafic global contient de nombreux flots de débit crête inférieur à p, l'approximation Poissonnienne peut être trop conservatrice. C'est pourquoi nous proposons un algorithme plus raffiné, dénoté (MinVar) pour *Minimum de Variance*, où l'estimation de la variance est le minimum de $B_t p$ et de celle calculée par (6.5). Les algorithmes (Poisson) (MinVar) sont évalués pour différents scénarios de test dans la section 6.5 ci-dessous.

6.4.4 Limite du nombre d'arrivées par intervalle

Le contrôle d'admission est particulièrement utile lors d'événements exceptionnels quand la demande dépasse considérablement la capacité du lien. Cela se produit en particulier lorsque une panne induit un reroutage du trafic sur le lien considéré. Le trafic va croître jusqu'à atteindre une nouvelle charge stationnaire. Le MBAC doit être capable de rejeter l'excédent de trafic dans ce régime afin de préserver les objectifs de performance. Cependant, l'impact majeur des pannes apparaît pour les flots en cours sur le lien coupé puisqu'ils apparaissent tous soudainement comme des nouveaux flots sur les liens de secours. Un lien sur ce chemin ne sera généralement pas congestionné avant la panne et donc disposé à accepter de nouveaux flots. Si toutefois tous les "nouveaux" flots qui arrivent dans les quelques intervalles de temps suivant la panne sont acceptés, le lien deviendra immédiatement fortement congestionné.

Pour prévenir cette situation, nous limitons le nombre de flots acceptés dans un intervalle de temps, comme dans [66]. Étant donnés les estimateurs A_t et $\widehat{\sigma}_t^2$, nous acceptons un maximum de n nouveaux flots où n est le plus petit entier tel que $(n+1)p + A_t + \alpha_q \widehat{\sigma}_t > C$. Il se peut que cela ne suffise pas vu le temps nécessaire avant que le trafic issu des nouveaux flots soit pris en compte dans l'estimation de la charge A_t. Il convient également de remarquer que, puisque les durées des flots ont généralement une distribution à queue lourde (*heavy-tailed*), l'espérance de la durée *résiduelle* des flots reroutés sera supérieure à la moyenne. L'impact des mauvaises décisions d'admission est ainsi plus sévère pour ces flots que pour ceux réellement nouveaux.

La stratégie de *back off* proposée dans [62] où, lorsqu'un flot est bloqué, aucun autre flot n'est accepté jusqu'à ce que l'un des flots en cours se termine, n'est pas applicable dans notre système puisque les terminaisons de flots ne sont pas explicites. La solution envisagée est d'interrompre la protection d'un nombre suffisant de flots en cours pour prévenir la congestion en remarquant que l'interruption des flots est dans tous les cas nécessaire quand la combinaison du trafic sur le lien en panne et le lien de secours dépasse la capacité disponible.

6.4.5 Instabilité du *priority load* mesuré

Le *priority load* peut varier abruptement lorsqu'un ensemble de flots de même débit P deviennent *backlogged* et qu'ils ne sont plus pris en compte dans sa mesure. Lorsque P est voisin du seuil p, il est possible qu'une situation anormale se produise où B_t est inférieur au seuil critique et la moyenne (à long terme) du *fair rate* reste au-dessus du seuil minimum, malgré le fait que des flots de débit p deviennent *backlogged*. Le lien sera alors ouvert à de nouvelles admissions. Ce phénomène sera d'autant plus marqué que le nombre de classes de débit est faible, puisqu'un grand nombre de flots sera rapidement sujet au basculement.

Afin de limiter l'impact de ce phénomène, nous imposons une valeur pour le *priority load* égale à $C\tau$ dans tous les intervalles ou un flot de débit crête p pourrait être *backlogged*. Cette dernière condition peut-être facilement déduite des paramètres de l'algorithme PFQ présenté dans le Chapitre 3. Il s'agit d'une mesure instantanée du *fair rate*, tandis que FR en est une moyenne sur le long terme.

Il n'est pas possible d'anticiper ces situations puisque dans Cross-Protect le *fair rate* instantané, qui cause le basculement, n'est connu qu'au moment où ce dernier se produit. Le contrôle exercé sur le *priority load* PL permet de limiter l'occurrence de tels instants, il demeure essentiel car l'indicateur sur le *fair rate* instantané seul ne serait pas stable (indication binaire).

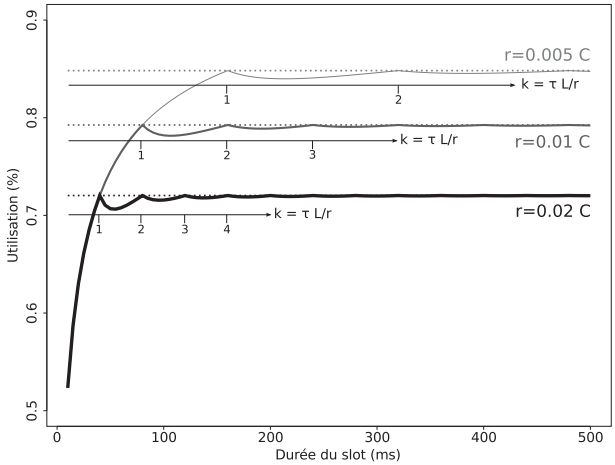


FIGURE 6.2 – Valeur du *priority load* mesuré en fonction de la taille de l'intervalle, pour différentes valeurs de p. La valeur réelle est indiquée par les lignes en pointillés.

Cette modification se montre également utile lorsque les flots ont un débit crête nominal p mais subissent de la gigue ; ces derniers se retrouvent momentanément *backlogged* quand l'intervalle entre deux arrivées de paquets est trop faible (voir la section 6.5.6).

6.4.6 Influence de la taille du slot sur les mesures

Le choix de l'intervalle de discrétisation sera généralement guidé soit par des contraintes techniques[2], soit avec l'objectif de mesurer un débit crête donné pour les flots. Nous considérons des intervalles de longueur kL/p pour $k \geq 1$. De plus petites valeurs de k ne permettent pas de prendre en compte la faible variance des flots de débit inférieur à p. D'autre part, de trop grandes valeurs de k ont tendance à rendre le MBAC moins réactif à des changements soudains tels que ceux considérés dans les scénarios de *flash-crowd*. Nous remarquons que les grandes valeurs de k n'apportent qu'une amélioration mineure de la performance. C'est pourquoi une valeur de quelques unités semble un bon compromis. Nous évaluerons l'impact de k dans la section 6.5.

Valeurs de k et estimation de la variance

Afin de comprendre l'impact du choix de cet intervalle sur l'estimation de la variance, nous considérons un flot émettant des paquets de taille L à débit constant sur un intervalle τ quelconque. Les calculs détaillés dans un annexe à ce chapitre donnent :

$$V_t = \frac{B_t}{p}\left[\frac{\Delta_t^2 L^2}{\tau^2} + (\frac{\tau p}{L} - \Delta_t)(1 + 2\Delta_t)\frac{L^2}{\tau^2}\right] \tag{6.8}$$

avec $\Delta_t = \lfloor \frac{\tau p}{L} \rfloor$

Puisque la variance est monotone en fonction de la charge, une itération jusqu'au point fixe permet de déterminer le seuil optimal de *priority load* pour ϵ donné. La figure 6.2 présente ces valeurs, pour $\epsilon = 10^{-2}$, et $p = 50$, 100 et 200kb/s.

Pour $k = 1$, la variance est celle du cas Poisson, qui permet d'estimer convenablement la variance et donc d'obtenir une utilisation optimale. La performance est sensiblement la même pour $k > 1$, et reste la même pour les valeurs entières de k. Par contre, pour $k < 1$, la variance est surestimée et coïncide avec la variance d'un flot de débit plus important. C'est-à-dire que si $\tau = L/s$ avec $r < s < p$, alors la performance atteinte sera celle correspondant au débit s et non pas r. La moyenne quant à elle n'est pas affectée. Par exemple, pour $p = 100$kb/s, une valeur de $k = 4$ permettra de mesurer correctement la variance de flots ayant un débit aussi faible que $r = 25$kb/s, et donc d'obtenir la performance correspondante. En ajustant le seuil d'admission à une valeur plus optimale, on s'attend à ce que le MBAC (MinVar) assure une meilleure différenciation des flots lorsque k est suffisamment grand.

2. C'est le cas par exemple du noyau Linux qui selon les versions présente une résolution minimale par défaut de 1 ou 10ms

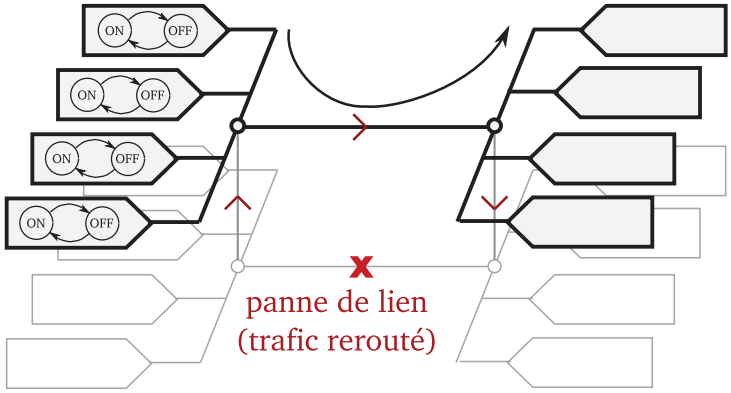


FIGURE 6.3 – Topologie utilisée pour la simulation : la partie représentée en trait gras est utilisée pour les simulations en régime stationnaire ; l'ensemble de la topologie sera utilisée pour simuler un scénario de *flash crowd* dans la section 6.5.7

.

6.5 Évaluation des algorithmes

Nous évaluons les propositions de MBAC (Poisson) et (MinVar) à l'aide d'un grand nombre de simulations, et nous comparons les résultats avec l'algorithme (GT) que nous avons présenté plus haut.
.

6.5.1 Environnement de simulation

La topologie pour la simulation est la topologie classique *dumbbell* avec un lien central de capacité $C = 10\text{Mb/s}$. Le choix délibéré d'une faible valeur de C permet d'obtenir des temps de simulation suffisamment courts, puisque la performance ne dépend que du ratio C/p (confirmé par simulations).

Le trafic est composé de flots UDP dont la durée est exponentiellement distribuée de moyenne $T_h = 60\text{s}$, et qui arrivent selon un processus de Poisson. Les flots génèrent leurs paquets selon un processus *on-off* assez général, de débit crête constant pendant la période *on* (de 50 à 300 kb/s) ; chaque période a une durée exponentielle de moyenne 500ms (sauf indication contraire). La taille des paquets est constante et fixée à 1000 octets. Les simulations sont lancées pendant 2000s plusieurs fois (25), et nous ne conservons que le régime stationnaire (en éliminant les 200 premières secondes). La probabilité de débordement cible ϵ est fixée à .01. Nous simulons une charge stationnaire égale à 100, 120 et 140% de la capacité du lien. Sauf indiqué, le taux d'arrivée des flots est tel que la limite d'admission par slot n'opère pas. L'intervalle $\tau = kL/p$ est choisi avec $k = 1$ ou une faible valeur indiquée. Nous notons qu'une valeur de p de l'ordre du seuil sur le *fair rate*, généralement choisi égal à 1% de C, correspondra à un pire cas pour la performance (voir tableau 6.1).

6.5.2 Critères de performance

La mesure de la performance des flots *streaming* et élastique nécessite de sélectionner des métriques qui reflètent la différentiation effectuée par notre ordonnanceur. La probabilité de débordement (notée *ov* pour *overflow*) représente la proportion d'intervalles où la charge prioritaire instantanée est supérieure à C. Elle est représentative de la performance des flots *streaming*, à condition que ceux-ci soient correctement envoyés dans la file prioritaire, ce qui est mesuré par la probabilité de *backlog* (*bk*, mesurée pour chaque classe de débit).

Nous mesurons également la probabilité de blocage (*bl*), qui représente la proportion de flots qui ne sont pas admis, ainsi que le taux de pertes de paquets (*lo* pour *losses*), la proportion de paquets qui sont perdus à cause de débordements du buffer. La combinaison des deux métriques montre comment l'excédent de trafic est éliminé pendant les instants de congestion. Un MBAC efficace devrait permettre une valeur de *bl* aussi faible que possible tout en garantissant la qualité de service des deux classes de flots. Nous détectons également le nombre de fois que chaque critère d'admission est responsable du blocage (bl_{PL} et bl_{FR}). Enfin, nous mesurons l'utilisation du lien (*ut*) qui, si elle dépend de la composition du trafic, nous donne cependant une indication de comment les ressources du lien sont exploitées.

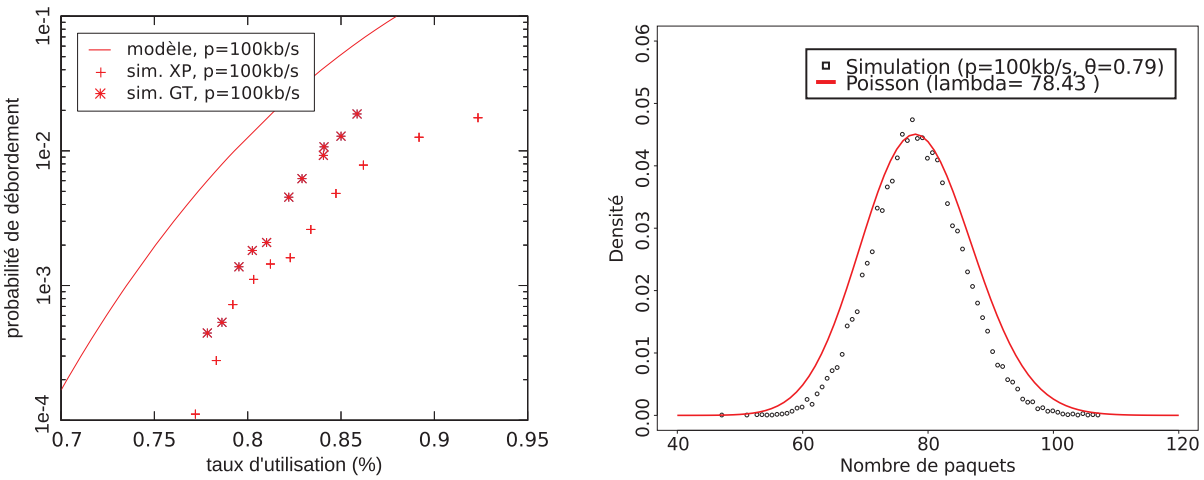


FIGURE 6.4 – A gauche : utilisation en fonction de la probabilité de débordement pour les MBAC XP et GT, comparés un cas où la charge par intervalle suit une distribution de Poisson, avec $p = 100\text{kb/s}$ – A droite : densité de charge reçue par intervalle pour $\rho = 1.20$, soumise au contrôle d'admission, comparé à une distribution de Poisson ajustée, pour $p = 100\text{kb/s}$.

6.5.3 Utilisation et probabilité de débordement

La relation entre l'utilisation et la probabilité de débordement représente la performance optimale qui pourrait être atteinte par l'algorithme dans des conditions stationnaires. Elle est obtenue pour chaque MBAC en faisant varier son critère d'admission. Pour (GT), nous faisons varier ϵ et notons l'utilisation et la probabilité de débordement réalisées. Pour les algorithmes de Cross-Protect (Poisson) et (MinVar), notés (XP) dans la figure, nous changeons simplement le seuil d'admission sur la charge B_t, sans utiliser ni la moyenne ni la variance. La figure 6.4 (à gauche) présente les résultats obtenus lorsque tous les flots ont un débit crête de 100kb/s. S'y trouve également la même relation pour une charge théorique suivant une distribution de Poisson.

Sans surprise au vu des résultats présentés dans [38], la performance obtenue est globalement la même pour les deux MBACs. Une différence subtile se produit à forte charge. Au fur et à mesure que le lien devient saturé, les flots dans Cross-Protect se retrouvent momentanément *backlogged* et ne contribuent plus au *priority load*, ce qui permet d'autres admissions et ainsi une utilisation plus élevée. De telles fortes valeurs du seuil correspondent à des situations où les flots de débit p deviennent *backlogged*, et où la performance n'est plus correctement mesurée par ov. Les seuils adaptatifs d'admission visent à prévenir de tels cas.

La figure suggère que la charge sur chaque intervalle est moins variable que Poisson, puisque l'utilisation atteinte est plus forte à probabilité de débordement égale. La figure 6.4 (à droite) montre l'histogramme empirique du nombre de paquets qui arrivent dans un intervalle pour des flots de débit crête 100kb/s sous le MBAC (Poisson). L'ajustement à une distribution de Poisson donne une nouvelle courbe proche de la mesure empirique, suggérant que notre approximation, certes conservatrice, reste très raisonnable (le trafic réel est moins variable). D'autres résultats non présentés ici montrent le même comportement pour d'autres valeurs de débit crête.

L'utilisation atteinte empiriquement ici pour un taux de débordement donné constitue une référence utile par la suite pour comparer la performance des deux variantes de notre algorithme. Le seuil d'admission optimal étant celui qui garantit l'utilisation la plus forte, tout en conservant une proportion négligeable de paquets traités en *backlog*.

6.5.4 Prédictibilité du MBAC XP

Bien que les algorithmes de MBAC possèdent généralement la même frontière de performance, ils diffèrent par la simplicité ou non de leur paramétrage, et par leur capacité à fournir une performance prévisible pour un jeu de paramètres donnés. Nous évaluons cette capacité pour les algorithmes (Poisson) et (MinVar) pour des flots de débit crête homogène r. Le débit protégé p peut être égal, supérieur ou inférieur à r. Des résultats avec un intervalle de confiance de 95% sont présentés dans le tableau 6.2 pour un intervalle $\tau = L/p$.

%	p	r	ov	ut	bl	bk	lo
Poisson	50	50	3.98e-4 ±0.92e-4	84.08 ±0.02	31.37 ±0.38	2.82e-3 ±0.52e-3	0.00 ±0.00
	100	100	3.11e-4 ±0.64e-4	78.44 ±0.06	35.20 ±0.61	8.14e-3 ±1.80e-3	0.00 ±0.00
	100	50	3.05e-3 ±0.20e-3	78.79 ±0.02	35.80 ±0.29	1.85e-5 ±2.19e-5	0.00 ±0.00
	100	300	2.71e-4 ±0.86e-4	97.63 ±0.96	3.04 ±1.49	76.27 ±3.32	15.60 ±1.14
MinVar	50	50	5.42e-3 ±0.31e-3	88.28 ±0.05	27.99 ±0.33	6.30e-2 ±0.44e-2	0.00 ±0.00
	100	100	3.24e-3 ±0.15e-3	83.59 ±0.08	30.98 ±0.59	1.32e-1 ±0.10e-1	0.00 ±0.00
	100	50	7.69e-3 ±0.30e-3	81.25 ±0.06	33.68 ±0.40	6.54e-4 ±5.62e-4	0.00 ±0.00
	100	300	2.13e-4 ±0.56e-4	98.33 ±0.69	1.84 ±0.98	78.91 ±2.56	16.21 ±1.08

TABLE 6.2 – Performance du MBAC pour Cross-Protect pour des flots de même débit crête

Flots de débit crête connu

Les résultats pour le cas $r = p$ sont présentés dans le tableau 6.2, pour $p = 50$Kb/s et $p = 100$kb/s. La probabilité de débordement reste d'un ou deux ordres de grandeur plus faible que notre objectif, ce qui conduit à une utilisation moins importante qu'il n'aurait été possible (déduite de la figure 6.4). Cependant, la perte en utilisation n'est pas trop importante et les contraintes de qualité de service sont respectées. Cette tendance conservatrice est un caractéristique commune à tous les MBAC. Les résultats pour le MBAC (MinVar) montrent que l'utilisation de la mesure de variance améliore la performance puisque l'agrégat de trafic est moins variant que Poisson (en raison du blocage notamment). La performance est encore meilleure pour $k = 2$.

Flots de débit crête plus faible

Généralement, la valeur de p sera suffisamment élevée pour protéger tous les flots *streaming*, et beaucoup auront un débit crête bien plus faible. Les lignes de la table 6.2 pour $p = 100$ et $r = 50$ montrent l'impact de supposer un débit crête plus élevé qu'il ne l'est. Avec le MBAC (Poisson), le seuil dépend uniquement de p de telle sorte que la performance réalisée avec $r = 50$kb/s est grossièrement similaire à $r = 100$kb/s. Cela illustre le coût d'utiliser le MBAC simple (Poisson), qui reste faible cependant avec une telle taille de lien (10Mb/s). (Minvar) est plus efficace dans cette situation lorsque l'intervalle d'échantillonnage est égal à kL/p et $k \geq 2$ (résultats non présentés ici) et, en fait, il donne la même performance que le MBAC (GT) (l'utilisation progresse de 81.25 $\pm$ 0.06 % à 88.02 $\pm$ 0.06 %).

Flots de débit crête plus important

Le dernier cas où $r > p$ illustre l'avantage majeur du MBAC (XP). Quand le critère d'admission est fixé à $p = 100$, les flots de débit crête $r = 300$ deviennent *backlogged* et ne contribuent plus que très faiblement au *priority load* (seulement les débuts de bursts). Le lien est complètement utilisé et ces flots perdent une forte proportion de leur paquets. La différenciation ne dépendant que du débit p, il est normal que l'on n'observe aucune différence entre les deux algorithmes. Enfin, dans les cas présentés, le *fair rate* à long terme ne passe jamais au dessous du seuil fixé à $C/100$ (ce qui correspond à une forte charge), d'où l'absence de blocage.

6.5.5 Performance avec un mélange de débits crête

Nous considérons maintenant deux classes de flots, de débits crête en dessous et au dessus de p, et une charge totale $1 \leq \rho_1 + \rho_2 \leq 1.40$. La figure 6.5 représente la différenciation réalisée dans un ensemble de simulations où $r_1 = 50$kb/s et $r_2 = 300$kb/s, respectivement avec (Poisson) (à gauche) et (MinVar) avec $k = 1$ (au milieu) et $k = 2$ (à droite). ρ_1 et ρ_2 sont représentées sur les axes x et y, et les nuances de gris représentent des niveaux de charge constante. Le symbole indique le niveau de différenciation : ✓ (totale), $\sim$ (bonne, se produit avec un certain délai), $\approx$ (différenciation épisodique), $\times$ (pas de différenciation).

Avec (Poisson), la différenciation se produit pour $\rho = 1.20$ et $\rho = 1.40$ dès que la charge de la classe de faible débit crête n'est pas trop importante. (MinVar) améliore la différenciation tout en maintenant les critères de QoS, notamment quand k est fixé de manière adéquate (ici $k = 2$). (MinVar) avec $k = 2$ parvient à différencier le trafic dans l'ensemble des configuration avec $\rho = 1.20$ et $\rho = 1.40$.

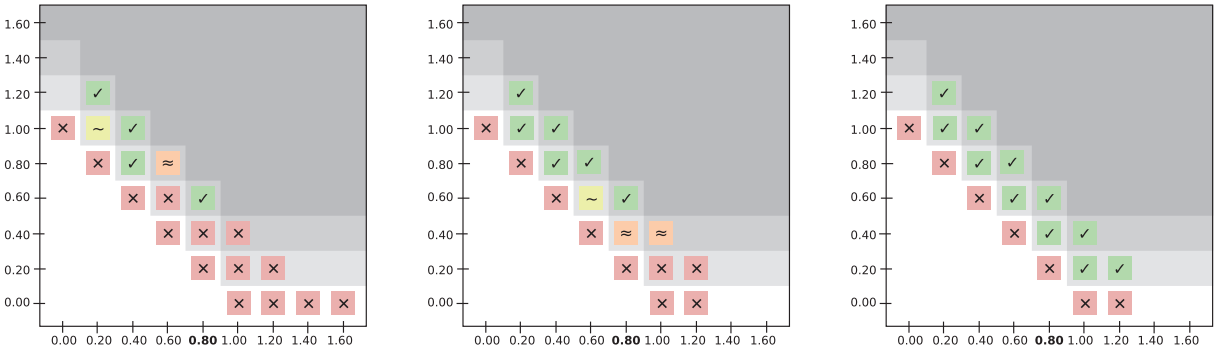


FIGURE 6.5 – Différenciation réalisée par (Poisson) avec $k = 1$ (à gauche), et (MinVar) avec $k = 1$ (au milieu) et $k = 2$ (à droite). Dans chaque figure, le débit crête protégé est $p = 100$kb/s, et la charge des flots de classe 1 ($r_1 = 50$kb/s) et 2 ($r_2 = 300$kb/s) est représentée respectivement en abscisse et en ordonnée.

En regardant de plus près les métriques de performance, nous voyons que la différenciation est variable selon les paramètres de trafic. La discrimination est reflétée dans les différentes valeurs du taux de blocage bl et du taux de pertes lo. Il est nécessaire dans tous les cas d'éliminer au moins l'excédent de charge ($bl + lo > (\rho - 1)/\rho$). Dans les cas sans discrimination, cela est réalisé uniquement par blocage, et l'utilisation reste relativement peu élevée. Dans les autres, les flots de plus fort débit perdent des paquets, ce qui réduit la proportion bloquée (qui est la même pour les deux classes) et l'utilisation est proche de 100%. Les flots différenciés sont servis en fonction de la bande passante laissée disponible par les flots prioritaires, et leur qualité de service est assurée par la borne inférieure sur le *fair rate* à long terme.

Le début de la simulation correspond à un régime transitoire où la charge prioritaire inclue l'ensemble du trafic entrant. Lorsque le lien commence à être saturé, les flots de plus fort débit crête deviennent *backlogged* et cessent de contribuer à cet estimateur. Cette situation est due au fait que le MBAC se base sur une estimation de la variance donnée par l'approximation Poissonnienne. Une fois la différenciation effectuée, PL décroit jusqu'à une valeur qui permet de nouvelles arrivées. Ce processus continue jusqu'à ce que le MBAC n'accepte plus aucun flot (lorsque le *fair rate* instantané deviendrait plus faible que p).

Parfois, la composition du trafic est telle que le MBAC ne permet pas que le lien devienne saturé. Cela se produit généralement quand l'estimation de la variance est basse à cause à la fois de débits crête relativement faibles, et d'une faible demande de la part des flots de fort débit. Les cas où p est petit comparé au seuil sur FR seront favorables à une différenciation. Dans tous les cas, nous voyons que les flots de débit crête inférieur à p sont protégés, puisqu'ils subissent des taux de débordement et de *backlog* négligeables, et aucune perte.

6.5.6 Impact de la gigue

En pratique, les flots de débit nominal r sont sujets à des délais variables et acquièrent de la gigue lorsqu'ils sont multiplexés dans les files d'attente successives des routeurs qu'ils traversent. Il est important de comprendre comment ce phénomène peut affecter la performance du MBAC. Une hypothèse de pire cas, selon la conjecture dite de "gigue négligeable" [33], est que les flots envoient des paquets selon un processus de Poisson de débit r pendant leurs périodes d'activité. **Cela sera vraisemblablement pire que la gigue réellement acquise par les flots**, notamment dans un réseau équipé de routeurs Cross-Protect où le *fair queueing* tend à restaurer l'espacement original des flots gigués. L'évaluation du MBAC soumise à un trafic Poissonnien au niveau paquet n'en reste pas moins intéressante en soi.

Nous avons ré-évalué la performance des deux algorithmes dans les scénarios précédents lorsque les flots ont acquis de la gigue. Pour des raisons de place, nous ne détaillons ici que nos principales observations.

Lorsque le débit des flots $r < p$, la gigue n'est pas assez importante pour avoir un impact sur la performance. La différence se produit lorsque r est proche de p. Les paquets de flots gigués sont temporairement retardés par l'ordonnanceur lorsque leur débit instantané est plus grand que p. Cela permet au MBAC de saturer le lien lorsque les caractéristiques du trafic le permettent. Les nouveaux flots sont bloqués par la condition sur FR, qui est un signe de forte charge. Alors que (GT) n'opère que sur l'ensemble du trafic, Cross-Protect peut mettre en attente les paquets qui possèdent un fort débit instantané, admettre de nouveaux flots et ainsi diminuer la probabilité de blocage. De cette façon, il

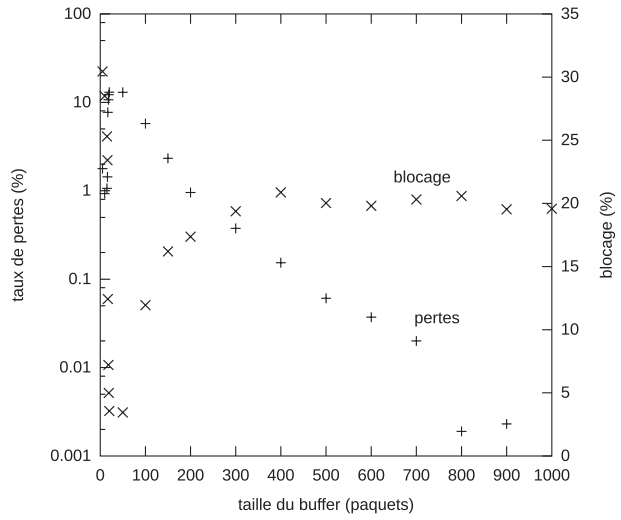


FIGURE 6.6 – Illustration du compromis taux de pertes de paquets / taux de blocage des flots en fonction de la taille du buffer pour $r = p = 100$kb/s

réduit la gigue, ce qui est un avantage supplémentaire de la différenciation. Il convient toutefois de s'assurer que le buffer est suffisamment dimensionné pour accueillir les paquets sans provoquer de perte.

La figure 6.6 présente à titre illustratif ce compromis pertes/blocage pour différentes valeurs de B, dans le cas $r = p = \theta_{FR} = 100$kb/s qui est le plus défavorable ici.

6.5.7 Contrôle d'admission en régime non-stationnaire : situations de *flashcrowd*

Nous considérons maintenant les cas non-stationnaires introduits dans la Section 6.4.4, en limitant notre étude à un scénario simple où tous les flots sont UDP[3]. Les flots ont les mêmes propriétés que dans le cas homogène précédent (arrivées selon un processus *on/off*, distribution exponentielle de leur taille). La topologie simulée est illustrée en figure 6.3 et consiste en deux topologies parallèles similaires aux simulations précédentes. La charge initiale sur chacune est de 60%. La durée de simulation est de 1000s ; le second lien central subit une panne à $t = 500$s, ce qui cause le reroutage de tout le trafic sur le premier lien qui devient goulotté.

Nous fixons $p = 100kb/s$ et considérons des flots de débit crête $r_a = 100$kb/s (un pire cas pour la performance). La figure 6.7 montre l'évolution des estimateurs FR (à gauche) et PL (à droite), accompagnés de leurs valeurs instantanées. Après la panne, les flots de débit r_a deviennent *backlogged* le temps de quelques secondes jusqu'à ce que le lien retrouve le régime stationnaire tel qu'obtenu dans la section 6.5.4. Le *fair rate* instantané devient inférieur à p, ce qui indique que trop de flots ont été admis. Pendant l'événement de *flash-crowd*, le blocage des flots est dû uniquement à la limite du nombre d'admissions introduite dans la section 6.4.4. L'ajout de la valeur du débit protégé p à la charge mesurée B_t permet de corriger l'estimateur qui est ainsi mis à jour rapidement malgré le coefficient de lissage. Il serait inutile sinon puisque l'échelle de temps caractéristique calculée n'est plus représentative du taux d'arrivées et de départ des flots. Nos simulations montrent que la performance est bien plus affectée si nous désactivons cette condition supplémentaire.

Afin de comprendre l'impact du débit crête sur la performance des flots *streaming*, nous considérons également le cas $r_b = 50$kb/s. La figure 6.8 trace la probabilité de *backlog* avec des flots de débit r_a (à gauche) and r_b (à droite), avec les deux algorithmes et dans un ensemble de configurations. Pendant quelques secondes après le reroutage, une grande partie des flots *streaming* est *backlogged*. Avec r_b, ce taux est bien plus faible, ce qui est encourageant puisque comme nous l'avons déjà précisé, la valeur de p sera généralement bien plus élevée que le débit effectif des flots à protéger.

Bien qu'encourageants, ces résultats préliminaires illustrent clairement la difficulté de contrôler le trafic au travers d'un mécanisme de contrôle d'admission, dans des cas tels que celui considéré. La condition limitant le nombre de flots acceptés dans un intervalle de temps peut s'avérer trop conservatrice lorsque les flots ont un débit crête inférieur à p ou, comme observé dans des traces réelles, s'il y a de nombreux flots consistant en un seul paquet. Le blocage des nouvelles arrivées après avoir détecté une congestion comme nous le faisons ici peut être insuffisant pour des régimes non-stationnaires.

3. Les flots TCP reroutés pourraient se retrouver en mode *slow start*, et accroître leur débit progressivement après avoir été acceptés, ce qui combiné avec leur contrôle adaptatif, rend la performance moins compréhensible.

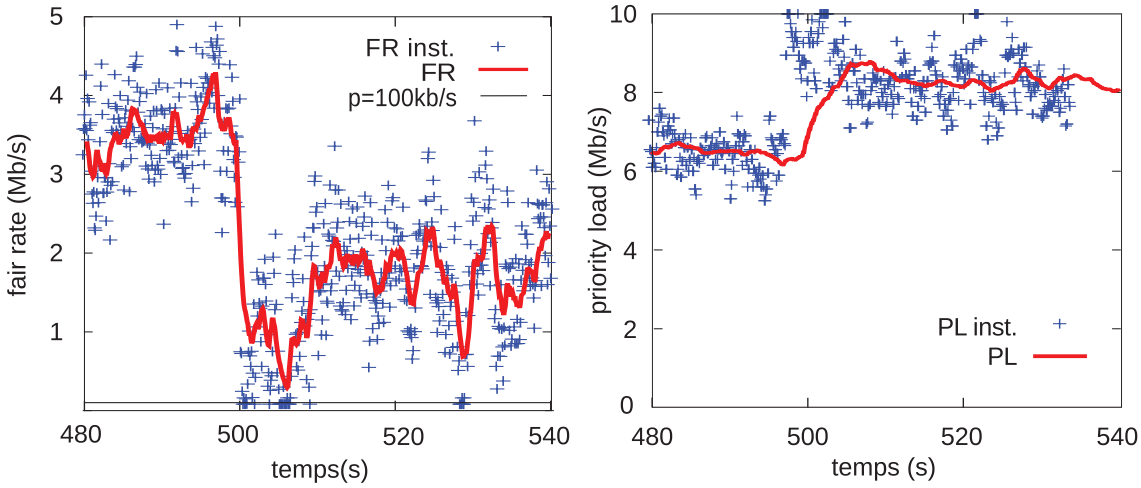


FIGURE 6.7 – Évolution du *fair rate* (à gauche) et du *priority load* (à droite) lors d'un événement de *flashcrowd* avec MinVar, $k = 2$, $p = 100\text{kb/s}$ et $r = 100\text{kb/s}$

Cette remarque est d'autant plus justifiée si l'on considère des scénarios plus réalistes avec des flots TCP et des distributions à queue lourde. Des travaux futurs peuvent considérer l'ajustement des facteurs de lissage en fonction du changement de la variance qui est caractéristique de telles situations. Il est également possible de tenter d'utiliser des algorithmes de détection de changement (issus de la théorie du signal). La condition est que ces solutions restent relativement simples à implémenter sur des liens à haut débit. Il faut cependant garder à l'esprit que des mesures plus radicales seront sans doute nécessaires pour empêcher une dégradation globale de la performance, comme l'interruption de flots déjà établis.

6.6 Conclusions

La proposition FAN basée sur les mécanismes Cross-Protect permet des garanties de performance pour les flots *streaming* et élastiques sans la complication de devoir marquer les paquets pour une différenciation de service. Il est toutefois nécessaire de proposer et calibrer des algorithmes de contrôle d'admission adaptés. Des travaux précédents suggèrent qu'un algorithme simple fondé sur une estimation du *fair rate* est suffisant lorsque la plupart des flots constituant le trafic sont élastiques et goulottés sur le lien considéré. Cependant, dans un réseau de cœur, la majorité des flots est contrainte ailleurs, par exemple par leurs débits d'accès plus faible que celui du lien. Le trafic est alors composé d'un ensemble de flots avec des débits crête limités. Certains ont un débit inférieur au *débit protégé* p (et ils doivent être servis en priorité), tandis que pour d'autres sont de débit supérieur à p. Le challenge est de proposer un MBAC qui permette à ces derniers de devenir *backlogged* (ils sont supposés à débit adaptatif), tout en continuant de servir les autres en priorité. Cela est d'autant plus délicat que l'algorithme ne peut se baser sur aucune connaissance des caractéristiques des flots.

Dans ce chapitre, nous avons proposé un MBAC simple basé sur des mesures de la moyenne et de la variance de la charge offerte à la file prioritaire de Cross-Protect. Il diffère de celui proposé par Grossglauser et Tse [66] en ce que la variance n'est utilisée que si elle est inférieure à la variance estimée lorsque l'on suppose que tous les flots ont le débit cible p. Nos simulations montrent que notre proposition permet la différenciation que nous recherchons dès lors que les débits des flots ne sont pas trop proches les uns des autres.

Les évaluations dans un scénario de *flash-crowd* sont moins encourageantes. Il est difficile pour de nombreuses raisons de trouver un compromis acceptable entre un trop grand nombre de flots acceptés – qui entraîne une période importante où la performance est dégradée – et un fort conservatisme qui se traduit par le rejet de plus de flots que nécessaire. Une étude plus approfondie d'un tel scénario est nécessaire vu son importance pratique. Nos résultats suggèrent qu'un contrôle d'admission simple n'est peut-être pas suffisant. Il est possible que l'interruption d'un sous ensemble des flots en cours soit également nécessaire afin de retrouver un niveau de charge acceptable.

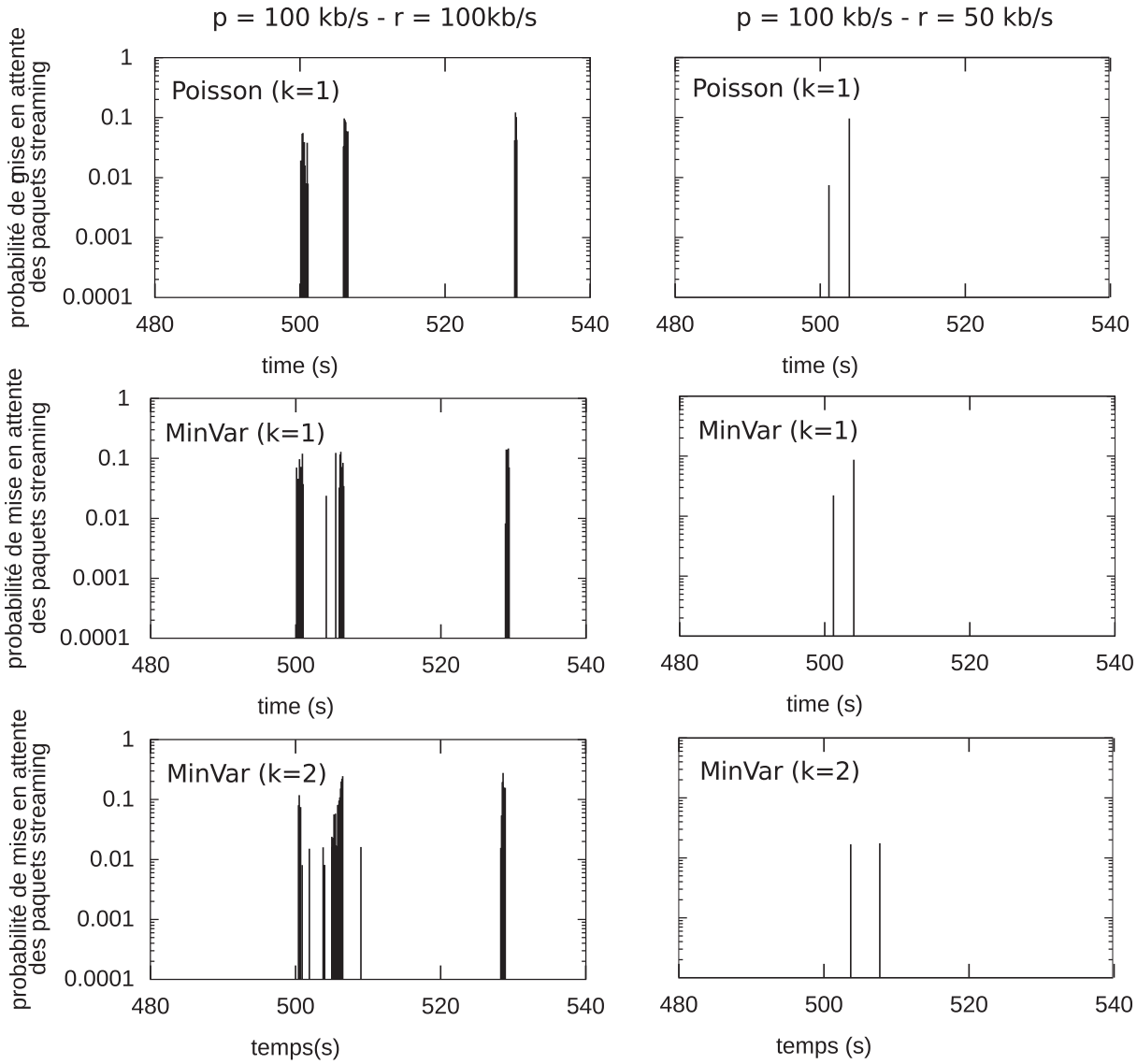


FIGURE 6.8 – Probabilité de mettre des paquets *streaming* en file d'attente dans deux scénarios : $p = 100$kb/s, $r = 100$kb/s (à gauche) et $p = 100$kb/s, $r = 50$kb/s (à droite), avec Poisson, $k = 1$ (en haut), MinVar, $k = 1$ (au milieu), and $k = 2$ (en bas).

6.A Estimation de la variance sur un intervalle de taille quelconque

Afin d'obtenir le résultat souhaité, nous devons d'abord établir quelle est la variance d'une somme d'un nombre aléatoire de variables aléatoires.

$$\mathrm{Var}\left[\sum_{i=1}^{N_t} M^{(i)}\right] = \mathrm{Exp}[S^2] - \mathrm{Exp}[S]^2, \text{ avec } S = \sum_{i=1} N_t M^{(i)}$$

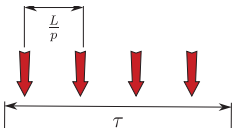
$$\mathrm{Exp}[S] = \mathrm{Exp}[N] \cdot \mathrm{Exp}[M]$$

$$\begin{aligned}
\mathrm{Exp}[S^2] &= \mathrm{Exp}\left[\sum_{i=1}^{N_t} M_i^2 + \sum_{i=1, i\neq j}^{N_t}\sum_{j=i}^{N_t} M_i M_j\right] \\
&= \mathrm{Exp}[N]\,\mathrm{Exp}[M_i^2] + \mathrm{Exp}\left[N(N-1)\,\mathrm{Exp}[M]^2\right] \\
&= \mathrm{Exp}[N]\,\mathrm{Exp}[M^2] + \left[\mathrm{Exp}[N^2] - \mathrm{Exp}[N]\right]\mathrm{Exp}[M]^2
\end{aligned} \tag{6.9}$$

D'où :

$$\begin{aligned}
\mathrm{Var}[S] &= \mathrm{Exp}[N]\,\mathrm{Exp}[M^2] + \mathrm{Exp}[N^2]\,\mathrm{Exp}[M]^2 - \mathrm{Exp}[N]\,\mathrm{Exp}[M]^2 - \mathrm{Exp}[N]^2\,\mathrm{Exp}[M]^2 \\
&= \mathrm{Exp}[N]\,\mathrm{Var}[M] + \mathrm{Var}[N]\,\mathrm{Exp}[M]^2
\end{aligned} \tag{6.10}$$

Nous déterminons maintenant la variance mesurée sur un intervalle de taille τ quelconque, en supposant toujours que les flots ont un débit crête constant p pendant leur période d'activité :



Puisque nous avons un processus d'arrivée Poissonnien, le nombre de flots actifs dans l'intervalle t, N_t, est également Poissonnien, au taux λ. Nous appliquons également une hypothèse de séparation des échelles de temps afin de pouvoir considérer que le nombre de flots actifs dans un intervalle est constant.

Le nombre de paquets reçus pour un flot actif pendant l'intervalle t est $M_t = D_t + R_t$ où $D_t = \lfloor \frac{\tau p}{L} \rfloor$ et R_t tel que $\Pr\{R_t = 0\} = \frac{\tau p}{L} - D$ et $\Pr\{R_t = 1\} = 1 - \frac{\tau p}{L} + D$.

$$\begin{aligned}
A_t &= \mathrm{Exp}[\sum_{i=1}^{N_t} M_t^{(i)}]\frac{L}{\tau} \\
&= \mathrm{Exp}[N_t] \cdot \mathrm{Exp}[M_t]\frac{L}{\tau} \\
&= \lambda\left[(D_t+1)(\frac{\tau p}{L} - D_t) + D_t(1 - \frac{\tau p}{L} + D_t)\right]\frac{L}{\tau}) \\
&= \lambda\left[\frac{\tau p}{L} - D_t\right]\frac{L}{\tau} \\
&= \lambda p
\end{aligned} \tag{6.11}$$

$$V_t = \left[\sum_{i=1}^{N_t} M_t^{(i)}\right]\frac{L^2}{\tau^2} \tag{6.12}$$

Nous avons alors :

$$\mathrm{Exp}[M_t] = D_t + \frac{\tau p}{L} - D_t = D_t + \gamma \tag{6.13}$$

$$\mathrm{Var}[M_t] = \mathrm{Var}[R_t] = \left(\frac{\tau p}{L} - D_t\right)\left(1 - \frac{\tau p}{L} + D_t\right) = \gamma(1-\gamma) \tag{6.14}$$

En insérant 6.13 et 6.14 dans 6.10, nous obtenons :

$$
\begin{aligned}
V_t &= \lambda \left[\gamma(1-\gamma) + (D_t + \gamma)^2 \right] \frac{L^2}{\tau^2} \\
&= \lambda \left[\frac{D^2 L^2}{\tau^2} + \gamma(1 + 2D_t) \frac{L^2}{\tau^2} \right] \\
&= \frac{A_t}{p} \left[\frac{D^2 L^2}{\tau^2} + (\frac{\tau p}{L} - D_t)(1 + 2D_t) \frac{L^2}{\tau^2} \right] \qquad (6.15)
\end{aligned}
$$

Quand $p < \frac{L}{\tau}$, $D_t = 0$ et V_t se simplifie en :

$$
\begin{aligned}
V_t &= \lambda \frac{\tau p}{L} \cdot \frac{L^2}{\tau^2} \\
&= \lambda \frac{PL}{\tau} \\
&= A_t \cdot \frac{L}{\tau} \qquad (6.16)
\end{aligned}
$$

Si $\tau \leq \frac{L}{p}$ alors la mesure de variance reste égale à la mesure obtenue dans le cas simpliste où $\tau = \frac{L}{p}$ que nous avons vu précédemment.