

Contrôle de débit TCP-compatible pour la vidéo sur l'Internet

Jérôme Viéron et Christine Guillemot*
IRISA/INRIA Rennes
Campus universitaires de Beaulieu
35042 Rennes Cedex
jerome.vieron@irisa.fr

Résumé

Ce papier décrit un algorithme de contrôle de débit prenant conjointement en compte les contraintes de compatibilité avec TCP ainsi que les contraintes de délai des flux multimédia temps réel. Etant donné l'état du réseau ainsi que les états des buffers d'émission et de réception, les contraintes temps-réel de la vidéo sont traduites en une contrainte de débit pour l'encodeur. Le modèle global de contrôle de débit proposé appliqué à une source vidéo permet de réduire de manière significative les pertes dues aux retards et donc de minimiser la distorsion tout en maximisant l'utilisation de la bande passante TCP-compatible.

Mots Clef

Internet, vidéo, temps-réel, débit, TCP-compatible

1 Introduction

Les besoins en terme de QoS des flux multimédia diffèrent de ceux des communications informatiques classiques. Les variations de délai de bout-en-bout ont un impact plus important sur les flux continus que sur les communications traditionnelles. Il est donc nécessaire de concevoir des nouvelles stratégies de contrôle de congestion, dédiées à ces flux continus, au moins aussi réactives que TCP afin de maintenir une certaine équité entre les échanges de données traditionnelles et les transmissions multimédia. Différents modèles dits TCP-compatibles ont été proposés dans la littérature [1,2,3]. Toutefois, ceux-ci ne prennent pas en compte les caractéristiques des flux multimédia temps-réel transmis (délai de bout-en-bout, paquets de taille variable, etc.). De plus, les expérimentations montrent que l'approche classique, qui consiste à réintroduire directement la valeur de la bande passante prédite comme contrainte pour le codeur vidéo, peut mener à un nombre important de retards et donc de pertes dues aux contraintes temps-réel de la source.

Ce papier décrit un algorithme de contrôle de débit TCP-compatible d'une source vidéo fondé sur le pro-

tolele TFRC [3] et construit au dessus du protocole RTP (Real-time Transport Protocol).

Le schéma global de contrôle de débit proposé prend en compte les modèles de délai et de buffers de la source afin de minimiser la distorsion au récepteur. Ces performances ont été, de plus, améliorées en y intégrant une nouvelle stratégie de diminution de la fréquence temporelle de la source visant à rendre la qualité visuelle du rendu la plus constante possible. Le modèle global proposé permet de réduire de manière significative les pertes dues aux retards et donc de minimiser la distorsion tout en maximisant l'utilisation de la bande passante TCP-compatible.

2 Protocole TCP-compatible

Le protocole TFRC (TCP-Friendly Rate Control) [3] est basé sur le modèle de débit TCP proposé dans [2],

$$B = \frac{MSS}{RTT \sqrt{\frac{2bp}{3}} + T_o \min(1, 3\sqrt{\frac{3bp}{8}}) p (1 + 32p^2)}, \quad (1)$$

où MSS représente la taille maximale d'un paquet, RTT est le délai d'aller-retour entre la source et le récepteur, p est le taux d'événements de pertes. Le paramètre b représente le nombre de paquets acquittés par un rapport de réception et T_o le délai avant retransmission d'un paquet TCP.

Ce protocole a été modifié et adapté afin de prendre en compte pleinement les caractéristiques des flux multimédia et de fonctionner au dessus de RTP. Les différents paramètres sont estimés via des rapports RTCP (Real-time Transport Control Protocol) entre le récepteur et la source.

2.1 Fréquence des rapports

L'adaptation de débit d'une source vidéo se fait en règle générale à certains instants clés du processus d'encodage. C'est pourquoi nous avons choisi ici de synchroniser la réception des rapports de réception avec la frontière d'une image tout en respectant les recommandations d'utilisation de la bande passante

*Ce travail est mené dans le cadre du projet RNRT VISI.

réservée à RTCP. Ainsi, la fréquence δ_{feed} entre deux rapports est définie comme suit,

$$\delta_{feed} = \frac{[\max(RTT, \frac{Avgsize \times NR}{0.75 \times 0.05 \times SB}) \times FR]}{FR},$$

avec NR le nombre de récepteurs dans la session et SB le débit total alloué à la session. Le terme $Avgsize$ représente la taille moyenne des rapports RTCP. Le RTT est calculé en utilisant une moyenne exponentielle pondérée avec un facteur de lissage de 0.9.

2.2 Événements de congestion

La principale modification apportée à TFRC réside dans l'intégration de la notion de paquets de taille variable dans le calcul du taux d'événements de pertes p . C'est pourquoi, la notion d'événements de pertes est révisée afin de considérer la quantité réelle de données perdues. Soit n le nombre d'intervalles de temps qui compose la fenêtre d'observation. Notons $SizeLost(i)$ la quantité cumulée de données perdues (en octets) sur l'intervalle de temps i . Alors, le nombre d'événements de pertes c_i (en unité de MSS) est donné par

$$c_i = \min(1, \frac{SizeLost(i)}{MSS}).$$

Comme dans [3], afin de rendre l'algorithme réactif tout en évitant des phénomènes d'oscillations importants, la bande passante prédite B est calculée sur une période de n intervalles de pertes (i.e. délai entre 2 pertes) et pondérée par un facteur d_i . Soit τ_i représentant la quantité totale de données transmises (en unité de MSS) pendant l'intervalle de pertes i , le taux d'événements de pertes s'exprime alors $p = \min(\hat{p}_{new}, \hat{p}_{old})$, avec

$$\hat{p}_{old} = \frac{\sum_{i=1}^n d_i w_i c_i}{\sum_{i=1}^n d_i w_i \tau_i} \quad \text{et} \quad \hat{p}_{new} = \frac{\sum_{i=1}^n d_{i-1} w_i c_i}{\sum_{i=0}^n d_i w_{i+1} \tau_i}.$$

3 Régulation de la source vidéo

Une fois la bande passante utilisable estimée, le problème consiste à transcrire cette bande passante en une contrainte de débit pour le codeur vidéo.

3.1 Approche directe

Une approche classique et simpliste consiste à réintroduire directement la bande passante estimée $R_N(t)$ comme contrainte de débit pour le codeur vidéo $R_e(t)$. Notons δ_{feed} le délai entre deux rapports de réception et t_n l'instant de réception d'un rapport. Alors, le budget de débit alloué au codeur, en utilisant ce modèle dit "Direct", sur l'intervalle de temps $[t_n, t_n + \delta_{feed}]$ peut s'exprimer comme suit,

$$\int_{t_n}^{t_n + \delta_{feed}} R_e(t) dt \leq \int_{t_n}^{t_n + \delta_{feed}} R_N(t) dt.$$

Ce budget de débit peut alors être facilement traduit en une contrainte de débit R_e^i pour chaque image $i \in \{n, n+1, \dots\}$. Cependant, comme le montre les expérimentations en section 5, ce type d'approche ne prenant pas en compte les contraintes de délais, ainsi que le niveau de remplissage des buffers d'émission et de réception peut mener à un nombre important de retards et donc de pertes de paquets influant de manière négative sur la qualité de la vidéo reçue.

3.2 Modèle global

Nous avons, donc, défini un modèle "Global" plus adapté et basé sur les différents éléments de la chaîne de transmission illustrée sur la (Fig.1).

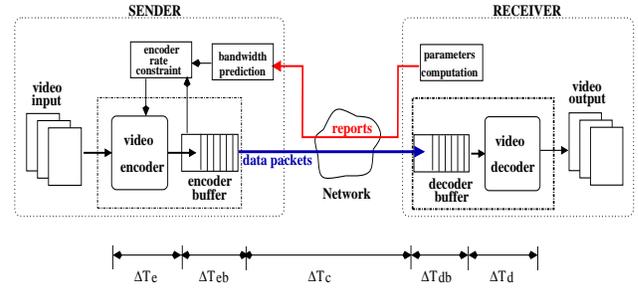


FIG. 1 – Chaîne de transmission vidéo.

Soit ΔT le délai de bout-en-bout relatif à une image vidéo [4],

$$\Delta T = \Delta T_e + \Delta T_{eb} + \Delta T_c + \Delta T_{db} + \Delta T_d, \quad (2)$$

avec ΔT_e et ΔT_d les délais respectifs de codage et de décodage. Les termes ΔT_{eb} et ΔT_{db} représentent les temps de bufferisation à la source et au récepteur, et ΔT_c correspond au délai de transmission. Afin de satisfaire aux contraintes temps-réel, chaque image capturée à l'instant t_n doit être prête pour l'affichage à l'instant $t_n + \Delta T$, et, par conséquent, doit être transmise avant l'instant $t_n + \delta_{buff}$.

Le délai de bufferisation δ_{buff} peut être calculé par $\delta_{buff} = \Delta T_{eb} + \Delta T_{db} = \Delta T - \Delta T_e - \Delta T_c - \Delta T_d$. Ce délai représente le délai maximal (émetteur et récepteur compris) pendant lequel un fragment d'image pourra être bufferisé. Soit $\tilde{R}_N(t)$ la bande passante prédite alors le débit du codeur $R_e(t)$ est contraint comme suit :

$$B_e(t_n) + \int_{t_n}^{t_n + \delta_{feed}} R_e(t) dt \leq \int_{t_n}^{t_n + \delta_{feed} + \delta_{buff}} \tilde{R}_N(t) dt. \quad (3)$$

avec $B_e(t_n)$ le nombre de bits restant dans le buffer du codeur à l'instant t_n . Notons \tilde{R}_e^i le nombre de bit appartenant à la $i^{ème}$ image restant dans le buffer du codeur, et p le nombre d'image, présentes dans ce buffer à un instant donné t_n . $B_e(t_n)$ s'exprime de la manière

suivante :

$$B_e(t_n) = \sum_{i=1}^p \tilde{R}_e^i.$$

Afin de prendre en compte les délais de bufferisation, la contrainte de débit $\tilde{R}_N(t)$ est estimée sur l'intervalle $[t_n, t_n + \delta_{feed} + \delta_{buff}]$ en extrapolant la bande passante prédite sur l'intervalle $[t_n, t_n + \delta_{feed}]$. Du fait de la gigue au niveau du réseau le paramètre δ_{buff} devra être ré-estimé périodiquement comme décrit dans [5].

4 Adaptation de la source

L'algorithme précédent peut être amélioré en adaptant également le débit des images présentes dans le buffer d'émission. La procédure consiste à vérifier que chaque image présente dans le buffer ne sera pas en retard. Soit $(n-p)$ l'index de l'image la plus ancienne présente dans le buffer. La contrainte de débit pour l'image $(n-k)$ peut être exprimée par

$$\sum_{j=n-p}^{n-k} \tilde{R}_e^j \leq \int_{t_n}^{t_{n+1} + \frac{p-k}{FR} + \delta_{buff}} \tilde{R}_N(t) dt,$$

avec $k = p, \dots, 1$ et \tilde{R}_e^j représentant la quantité de données restantes relatives à la $i^{ème}$ image dans le buffer du codeur. Si la relation précédente n'est pas satisfaite le débit des images codées $(n-k)$ devra être réajusté selon la contrainte (3).

4.1 Fréquence temporelle

Nous avons, de plus, mis en oeuvre un nouvel algorithme de diminution de la fréquence temporelle de la source afin de limiter les variations en terme de qualité entre deux images consécutives dans un intervalle donné $[P_l - T_{qual}, P_l + T_{qual}]$ et ceci même en présence de contraintes de débits variables. Le terme P_l représente le PSNR de la dernière image codée et T_{qual} un seuil donné. Ainsi, le débit cible de la prochaine image \tilde{R}_e^n est contraint par

$$\tilde{R}_e^n(\tilde{D}_{qual-}) \leq \tilde{R}_e^n \leq \tilde{R}_e^n(\tilde{D}_{qual+})$$

avec $[\tilde{D}_{qual-}, \tilde{D}_{qual+}]$ représentant l'intervalle de distorsion visé (i.e. contraint par T_{qual}). Donc, si le budget de la prochaine image est inférieur à $\tilde{R}_e^n(\tilde{D}_{qual-})$ alors l'image n n'est pas codée. Si, au contraire, son budget est supérieur à $\tilde{R}_e^n(\tilde{D}_{qual+})$, nous contraignons son budget à cette dernière limite, afin de ne pas gaspiller inutilement la bande passante et d'augmenter la moyenne des débits des prochaines images à coder.

4.2 Modèle débit-distorsion

Les débits cibles \tilde{R}_e^n sont alors estimés par

$$\tilde{R}_e^i(\tilde{D}_j) = \sum_{i=1}^N B_i = A \sum_{i=1}^N \left(K \frac{\sigma_i^2}{12 \tilde{D}_j} + C \right),$$

avec B_j le coût en bits du $j^{ème}$ macrobloccs de l'image i , N le nombre de macrobloccs constituant une image, A le nombre de pixels dans un macrobloc (i.e. $A = 16^2$ pixels), K et C sont des constantes et σ_i est l'écart-type de la luminance et des chrominances du macrobloc i . La distorsion \tilde{D}_j est estimée par $\tilde{D}_j = Q_{mean}^2/12$, en considérant un pas de quantification moyen Q_{mean} pour chaque macrobloc de l'image j et un poids α_i égal à 1 pour chaque macrobloc. La méthode présentée ici se base sur les modèles de débit et de distorsion du codec H.263+ proposés dans [6].

5 Résultats expérimentaux

De nombreuses expérimentations ont été menées entre l'INRIA Rennes (France) et le laboratoire R.U.S (Allemagne) ainsi qu'entre l'INRIA Rennes et l'INRIA Sophia-antipolis (France), sur l'Internet. Ces expériences ont été réalisées afin de valider notre algorithme face à diverses configurations en terme de délais de transmission, de giges et conditions de pertes. Nous avons utilisé ici un codeur H.263+ implémentant la méthode de contrôle de débit du TMN8 modifiée pour s'adapter au débit variable. Le mode "Slice Structured" et l'option "Intra Forced Updating" ont été activés afin d'améliorer la robustesse du train binaire. La séquence vidéo *news* (CIF, 25 images/s) a été utilisée. Nous avons également pris un *MSS* de 1000 octets et un seuil T_{qual} de 1 dB.

Différents délais de bufferisation initiaux K , exprimés en nombre d'images, ont été testés (i.e. $K = 2, 3, 4$ images). K représente les délais cumulés de bufferisation à l'émetteur et au récepteur. Pour chacune des configurations, plus de 50 expériences ont été réalisées. Les résultats présentés ici (TAB 1) fournissent une synthèse des résultats obtenus entre l'INRIA Rennes et l'INRIA Sophia / R.U.S. Stuttgart. Le pourcentage d'utilisation (% bw) de la bande passante est calculé comme le ratio entre la quantité effective de données transmises et la valeur de bande passante utilisable.

Afin de tester le modèle dit "Direct", nous avons décidé d'expérimenter deux politiques de bufferisation initiale. Ainsi le modèle "Direct1" correspond à une bufferisation de $\sim 80\%$ de K du côté émetteur et $\sim 20\%$ du côté récepteur. A l'inverse le modèle "Direct2" correspond à une bufferisation de $\sim 20\%$ de K du côté émetteur et $\sim 80\%$ du côté récepteur. Dans le cas du modèle "Direct1", le nombre de paquets arrivés en retard est très important. De plus, ces retards, intervenant en rafales, ont un impact très négatif sur la qualité de la vidéo décodée. Ce phénomène peut être réduit en autorisant une plus forte bufferisation du côté du client, comme l'illustrent les résultats du modèle "Direct2". Cependant, les délais de bufferisation élevés sont incompatibles avec les applications nécessitant de l'interactivité.

Avec le modèle "Global", le nombre de pertes de pa-

Direct1	INRIA-Sophia			RUS-Stuttgart		
	K	2	3	4	2	3
% to	5.86	10.84	22.20	3.08	12.31	31.82
% bw	99.40	99.47	99.48	98.96	99.24	99.43
PSNR	35.27	34.08	31.59	36.51	33.99	28.80

Direct2	INRIA-Sophia			RUS-Stuttgart		
	K	2	3	4	2	3
% to	3.98	0.37	0.027	3.40	0.28	0.024
% bw	99.31	99.19	99.28	99.42	99.27	99.18
PSNR	36.93	37.09	37.10	37.11	37.10	37.16

Global	INRIA-Sophia			RUS-Stuttgart		
	K	2	3	4	2	3
% to	0.031	0.032	0.090	0.049	0.024	0.094
% bw	99.08	99.10	99.25	99.34	99.39	99.46
PSNR	37.41	37.45	37.49	37.41	37.34	37.28

TAB. 1 – Pourcentages de retards (% to), d'utilisation de bande passante (% bw) ainsi que le PSNR moyen obtenus entre Rennes et Sophia-Antipolis ($\delta_{feed} \simeq 80ms$) et entre Rennes et Stuttgart ($\delta_{feed} \simeq 120ms$) avec les modèles de contrôle de débit "Direct1", "Direct2" et "Global".

quets a été réduit de manière significative pour une utilisation comparable de la bande passante prédite. De plus, du fait des mécanismes relatifs à la fréquence temporelle de la source, la qualité de la vidéo obtenue reste relativement constante.

La figure 2 compare les valeurs de contraintes de débit fournies à l'encodeur, les taux d'événements de pertes ainsi que les PSNR du signal reçu pour les modèles "Direct1" et "Global" pour une valeur de $K = 3$. Ces courbes (représentatives des résultats obtenus) montrent que le PSNR dans le cas de l'approche "Direct1" est affecté de manière très significative par les retards des paquets. A l'inverse, avec le modèle "Global" ces retards sont assez réduits et leurs durées sont faibles. Ainsi, pour une utilisation comparable de la bande passante TCP-compatible prédite, les valeurs de PSNR obtenues pour la vidéo reconstruite sont plus élevées et plus stables.

6 Conclusion

Dans ce papier, nous avons décrit un nouveau protocole TCP-compatible, basé sur RTP, prenant en compte les caractéristiques des flux multimedia. De plus, nous avons développé un modèle global de régulation de débit qui considère à la fois les modèles de buffer et les contraintes de délai des flux temps-réel, pour la vidéo sur l'Internet. Le modèle proposé a été validé par un grand nombre d'expérimentations sur l'Internet [5]. Les performances ont, de plus, été améliorées par l'introduction d'une nouvelle stratégie de diminution de la fréquence temporelle des images qui permet d'obtenir

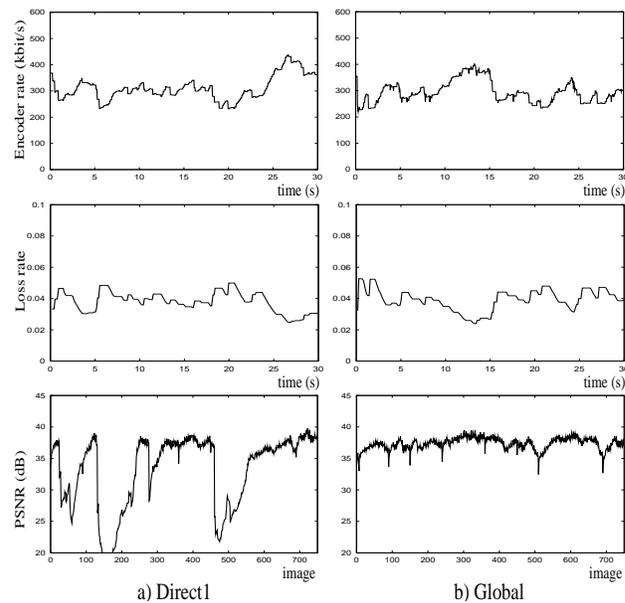


FIG. 2 – Contraintes de débits de l'encodeur, taux d'événements de pertes et courbes de PSNR pour les deux méthodes de contrôle de débit "Direct1" et "Global" ($K = 3$).

un meilleur compromis entre la fréquence temporelle et le PSNR obtenu. Des travaux en cours sont dédiés à la transposition du modèle proposé au cas d'une source vidéo dite Scalable à Grain Fin (FGS).

Références

- [1] J. Mahdavi and S. Floyd. TCP-friendly unicast rate-based flow control. Technical note sent to the end2end-interest mailing list, January 1997.
- [2] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose. Modeling TCP throughput: a simple model and its empirical validation. In *ACM SIGCOMM Symposium on Communications Architectures and Protocols*, August 1998.
- [3] S. Floyd, M. Handley, J. Padhye, and J. Widmer. Equation-based congestion control for unicast applications. In *Proceedings of ACM SIGCOMM'2000*, pages 43–56, May 2000.
- [4] C.Y. Hsu, A. Ortega, and M. Khansari. Rate-control for robust video transmission over burst-error wireless channels. *IEEE Journal on Selected Areas in Communications*, 17(5):756–772, May 1999.
- [5] J. Viéron and C. Guillemot. Real-time constrained tcp-compatible rate control for video over the internet. Submitted to *IEEE Trans. on Multimedia*, 2001.
- [6] J. Ribas-Corbera and S. Lei. Rate control for low-delay video communications. Technical report, ITU-Telecommunications Standardization, Video coding expert group, Doc Q15-A20, June 1997.