
Adaptation dynamique des débits des couches pour la transmission vidéo multipoint¹

Nicolas BONMARIAGE² — Guy LEDUC

Université de Liège
Research Unit in Networking
Institut Montefiore (B28)
B-4000 LIEGE 1
BELGIUM

{bonmarge,leduc}@run.montefiore.ulg.ac.be

RÉSUMÉ. La division du flux de données en couches cumulatives est proposée comme solution au problème de l'hétérogénéité de l'ensemble des récepteurs d'un flux vidéo multipoint. Un récepteur peut adapter à sa capacité le débit du flux reçu en s'abonnant à un nombre adéquat de couches. Cette approche statique n'est cependant pas suffisante pour garantir une utilisation judicieuse des ressources du réseau et doit être combinée avec un mécanisme d'adaptation dynamique des débits des couches. Nous étudions le problème de l'adaptation dynamique et développons une architecture intégrant un algorithme d'adaptation des débits et un protocole de transmission pour la vidéo multipoint en couches. Nous cherchons principalement à accommoder l'hétérogénéité des récepteurs tout en mettant en œuvre un contrôle de congestion et en garantissant l'équité vis-à-vis des flux TCP. La solution proposée résiste à un facteur d'échelle. Nous évaluons et analysons l'architecture développée au travers de simulations.

ABSTRACT. The heterogeneous environment of multicast video streams naturally raises a network performance issue which can be addressed using a layered transmission scheme. A receiver is able to adapt the flow rate to its capacity by choosing an adequate subscription level among the different layers. A traditional static approach doesn't allow a full usage of the network capacity and should be combined with an adaptive transmission rate in each layer. We study the dynamic rate adaptation problem and develop a framework integrating a multirate adaptation algorithm and a transmission protocol for layered multicast video. Our main focus is the development of an adaptation algorithm offering rates matching the capacities of the receivers. We also want to set up a congestion control scheme and offer fairness towards TCP flows, while keeping a scalable framework. We evaluate and analyze the proposed architecture using simulations to point out its main characteristics.

MOTS-CLÉS : multipoint, vidéo en couches, adaptation dynamique, contrôle de congestion, équité

KEYWORDS: multicast, layered video, dynamic adaptation, congestion control, fairness

1. Recherche partiellement financée par la Région Wallonne dans le cadre du projet TINTIN et par les SSTC dans le cadre du PAI MOTION.

2. Aspirant du Fonds National de la Recherche Scientifique (FNRS).

1. Introduction

La division en couches d'un flux de données est présentée comme solution au problème de l'hétérogénéité des récepteurs d'un groupe multipoint. Dans cette approche, les données d'un flux vidéo sont réparties entre couches successives en associant l'information de base aux couches basses et l'information de raffinement aux couches supérieures. Chaque couche étant émise à destination d'un groupe multipoint et à un débit spécifiques, la gestion par un récepteur de son abonnement à ces groupes lui permet d'adapter dans une certaine mesure le débit du flux reçu. L'information associée aux différentes couches est cumulative ; nous désignons dans ce contexte par *abonnement au niveau i* un ensemble de couches consécutives débutant à la couche de base et se terminant à la i^e couche.

La majorité des travaux dans ce domaine supposent pour la source une offre de débits statique, le nombre de couches et leurs débits ne connaissant aucune variation temporelle. Le problème de l'adaptation des débits des couches n'a pas été extensivement étudié.

Le premier véritable protocole de transmission multipoint en couches a été proposé dans [MCC 96]. Le protocole présenté dans cet article (*RLM*) ne réalise pas d'adaptation dynamique du débit des couches. Le contrôle de congestion est effectué par les récepteurs qui adaptent leur niveau d'abonnement sur base d'une estimation de la bande passante dont ils disposent. Le protocole RLM s'adapte relativement lentement aux variations de l'état de congestion du réseau et présente une redondance¹ non négligeable. Il n'offre de plus aucune garantie d'équité vis-à-vis des flux TCP.

Le protocole présenté dans [VIC 98] s'inscrit dans la lignée de RLM, tout en corrigeant certains défauts relatifs à la redondance du protocole ainsi que d'autres liés à l'absence de synchronisation des récepteurs. Le protocole offre en outre certaines garanties d'équité vis-à-vis des flux TCP. La solution proposée dans [El 01] s'inscrit dans cette lignée. Les garanties d'équité y sont toutefois plus fortes. Le protocole offre de plus une bonne dynamique et met en place un mécanisme d'apprentissage au niveau des récepteurs de manière à stabiliser leur point de fonctionnement.

Dans [SIS 00], les auteurs proposent une structure générale de protocole pour la transmission de flux multimédias multipoints en couches basé sur une adaptation dynamique des débits des couches. Les auteurs n'étudient cependant pas la problématique de l'adaptation en elle-même et ne proposent pas d'algorithme d'adaptation spécifique.

Un protocole complet pour la transmission multipoint en couches incorporant directement un algorithme d'adaptation dynamique des débits est proposé dans [VIC 00]. Les auteurs proposent deux architectures, l'une faisant intervenir activement les nœuds intermédiaires du réseau dans le cadre de la détermination des débits, l'autre se reposant uniquement sur les fonctionnalités implémentées au niveau de la source et des

1. Nous renvoyons le lecteur à [RUB 99] pour une définition formelle de la notion de redondance.

récepteurs. Les questions de l'évaluation des débits par les récepteurs et de l'équité ne sont cependant pas étudiées de manière très poussée.

L'approche considérée dans [LI 96] en réponse au problème de l'hétérogénéité consiste pour la source à émettre plusieurs flux distincts et non cumulatifs à des débits — et donc des qualités — différents. Quels que soient les mécanismes mis en œuvre pour limiter le coût en bande passante lié à l'utilisation de flux répliqués, la redondance présentée par cette approche sera toujours nettement plus élevée que celle liée à une transmission en couches. Le seul avantage réside dans l'utilisation d'un codeur vidéo plus simple. L'étude de l'adaptation dynamique des débits reste pertinente dans le cas de flux répliqués.

Nous développons dans cet article un algorithme d'adaptation dynamique des débits des couches. Afin d'en permettre la mise en œuvre, nous complétons celui-ci avec un protocole de transmission complet pour la vidéo multipoint en couches. Nous proposons au final une architecture modulaire permettant la mise en œuvre d'une adaptation dynamique des débits des couches par la source, de même que la mise en œuvre par les récepteurs d'un contrôle de congestion. Nous considérons également les problèmes de résistance à un facteur d'échelle et d'équité vis-à-vis des flux TCP.

L'article est organisé comme suit. Nous commençons par étudier le problème de l'adaptation dynamique des débits des couches de manière tout à fait générale. Nous poursuivons avec la présentation d'une architecture illustrant les possibilités de mise en œuvre conjointe d'une adaptation dynamique par la source et d'un contrôle de congestion par les récepteurs. Nous poursuivons avec une série de simulations illustrant les caractéristiques principales de cette architecture et concluons.

2. Adaptation dynamique des débits des couches

2.1. Définition du problème

Idéalement, chaque récepteur doit pouvoir trouver à tout instant un niveau d'abonnement coïncidant parfaitement avec le débit auquel il estime avoir droit. Nous supposons dans ce cadre que la source a la possibilité de modifier dynamiquement le nombre de couches émises de même que les débits associés à ces différentes couches.

Si la source cherche à offrir des débits correspondant aux attentes des récepteurs, l'information la plus pertinente sur laquelle l'adaptation peut se baser est bien évidemment l'ensemble des débits souhaités par ces derniers. La source ne peut cependant pas chercher à évaluer elle-même une information associée à l'ensemble de ses récepteurs. La charge associée serait incompatible avec les contraintes de résistance à un facteur d'échelle liées à l'environnement multipoint. Nous considérons qu'il est de la responsabilité de chaque récepteur de déterminer sa demande de débit et de l'acheminer à la source. Ne relevant pas directement de l'adaptation des débits, ces points sont envisagés plus loin. Nous supposons dans la suite de cette section que la source dispose comme information d'un ensemble de demandes de débits provenant des récepteurs.

Nous dérivons à partir de là une postcondition formelle que la source cherche à atteindre au terme de l'adaptation des débits. Soit $D = \{D_i\}$, $i = 1, \dots, m$, l'ensemble *ordonné* des demandes de débits parvenant à la source, soit n le nombre de couches émises par la source et soit L_i , $i = 1, \dots, n$, le débit *cumulé* au niveau i . Nous voulons déterminer une relation définissant les L_i , $1 \leq i \leq n$, en fonction des D_j , $1 \leq j \leq m$.

Idéalement, chaque récepteur devrait pouvoir trouver un niveau d'abonnement égal à sa demande ; la relation recherchée serait alors

$$L_i = D_i, \forall i \in [1, n] \quad (1)$$

avec $m = n$. Cette relation implique cependant que la source émette un nombre de couches égal au nombre de demandes reçues. Or, il est raisonnable de se fixer un nombre maximum n_{\max} de couches pouvant être diffusées par le codeur associé à la source et ce pour diverses raisons :

- la complexité du découpage du flux en couches et du réassemblage des couches risque en pratique de croître avec le nombre de couches ;
- chaque couche étant associée à un groupe multipoint, le nombre de groupes nécessaires à la diffusion du flux peut entraîner une consommation non négligeable d'adresses ainsi que d'espace mémoire au niveau des routeurs ;
- pour un débit maximum du codeur fixé, une augmentation du nombre de couches réduit le débit associé à chaque couche et augmente donc le surcoût lié aux en-têtes de paquets par rapport aux données.

Ces considérations introduisent une contrainte supplémentaire au problème qui est que le nombre n de couches émises par la source doit être inférieur ou égal à n_{\max} . La relation idéale (1) étant par conséquent limitée au cas où $m \leq n_{\max}$ (nous avons donc $n = \min(m, n_{\max})$), il est nécessaire de dériver une seconde postcondition tenant compte de cette contrainte et qui admette la première postcondition comme cas particulier lorsque $m \leq n_{\max}$.

Plaçons-nous dans le cas où la totalité des demandes ne peut être satisfaite. Le problème pour la source est alors de déterminer un ensemble de n_{\max} débits à émettre sur base d'un ensemble de demandes dont la cardinalité est supérieure à n_{\max} . Nous envisageons dans ce cadre une formulation générale de la postcondition de l'algorithme d'adaptation sous la forme d'un problème d'optimisation dans lequel l'adaptation de débits se réduit à la recherche de l'extremum d'une fonction scalaire mesurant la qualité de l'adaptation réalisée par la source, c'est-à-dire l'adéquation entre son offre de débits et les demandes provenant des récepteurs. Dans notre cas, nous considérons que les récepteurs cherchent à maximiser le débit qu'ils reçoivent sous contrainte que ce débit ne dépasse pas une valeur limite au-delà de laquelle de la congestion s'en suivrait dans le réseau. Chaque récepteur cherchera donc à s'abonner au plus haut niveau offert qui satisfasse la contrainte liée à l'absence de congestion.

La source doit sur cette base déterminer un ensemble de débits évitant que les contraintes associées aux récepteurs ne les forcent à opter pour un niveau d'abonnement qui soit trop inférieur à leur demande. C'est dans cette optique que nous définis-

sons l'offre optimale de débits pour la source : l'offre optimale est l'offre minimisant l'écart relatif moyen entre les demandes des récepteurs et les niveaux d'abonnement auxquels ils souscriront ensuite en vue de respecter leur contrainte de non-congestion.

Cette définition implique que le choix des débits offerts se fasse parmi l'ensemble $\{D_i\}$ des demandes des récepteurs. En effet, si l'ensemble des débits offerts n'était pas un sous-ensemble de $\{D_i\}$, nous pourrions toujours améliorer l'offre en remplaçant chaque débit offert par la demande immédiatement supérieure, ce qui permettrait à des récepteurs de se rabattre sur un débit plus élevé et fournirait donc un meilleur résultat.

Notons également que le débit L_1 de la couche de base est systématiquement fixé à D_1 afin de toujours satisfaire la plus petite demande. Cette opération peut avoir pour conséquence que l'offre de débits n'est pas optimale au sens où nous venons de le définir. Néanmoins, si nous avons $L_1 > D_1$, le récepteur associé à la plus petite demande n'aurait pas d'autre choix que de renoncer à la réception du flux émis par la source, sans quoi il violerait sa contrainte de non-congestion.

Le débit de la couche de base étant fixé, les débits des $n_{\max} - 1$ autres couches sont donc calculés de manière à minimiser l'écart relatif moyen entre une demande de débit donnée et le débit sur lequel le récepteur ayant émis cette demande doit finalement se rabattre. Formellement, si \bar{D}_i désigne le plus grand débit effectivement offert par la source qui soit inférieur ou égal à D_i , on cherche à déterminer $n_{\max} - 1$ autres débits tels que la somme

$$\frac{1}{\sum_{i=1}^m w_i} \sum_{i=1}^m w_i \frac{D_i - \bar{D}_i}{D_i} \quad (2)$$

soit minimale, ou encore, de manière équivalente, tels que la somme

$$\sum_{i=1}^m w_i \frac{\bar{D}_i}{D_i} \quad (3)$$

soit maximale, w_i représentant le nombre de récepteurs ayant requis le débit D_i .

La seconde postcondition énoncée ci-dessus admet la première comme cas particulier. En effet, lorsque la cardinalité m de l'ensemble de demandes $\{D_i\}$ est inférieure ou égale à n_{\max} , le sous-ensemble permettant de maximiser l'expression (3) est trivialement l'ensemble $\{D_i\}$ lui-même. Autrement dit, nous avons dans ce cas $L_i = \bar{D}_i = D_i$, $i = 1, \dots, m$.

Le problème d'optimisation discret (3) n'admet pas systématiquement une solution unique. Lorsque le choix entre plusieurs sous-ensembles optimaux se pose, nous sélectionnons le sous-ensemble correspondant aux demandes de débits les plus faibles et ce pour deux raisons. Premièrement, il en résulte une économie globale de ressources tant au niveau de la source que du réseau. Deuxièmement, nous pouvons supposer que le codage réalisé par la source est caractérisé par un taux d'accroissement de la qualité de l'image reçue qui diminue lorsque le débit du flux augmente. Pour un même niveau de satisfaction en termes de demandes des récepteurs, le choix d'un ensemble optimal

comportant des demandes plus faibles donnera lieu à une qualité moyenne de l'image plus élevée.

Formellement, si \mathcal{S}_m désigne l'ensemble des sous-ensembles ordonnés solutions du problème de maximisation pour un ensemble donné D de m demandes de débits, nous recherchons l'élément S_m^l de \mathcal{S}_m vérifiant la condition suivante :

$$(\neg \exists \bar{S} \in \mathcal{S}_m) (\exists k \in]1, \min(m, n_{\max})]) (\forall i \in [1, k]) \left[(s_{m_i}^l = \bar{s}_i) \wedge (s_{m_{k+1}}^l > \bar{s}_{k+1}) \right] \quad (4)$$

où, pour un ensemble A quelconque, la notation a_i désigne le i^e élément de cet ensemble. Si l'on considère les éléments de S_m^l un par un dans l'ordre croissant, la condition (4) signifie qu'il est impossible de leur substituer des demandes inférieures sans faire perdre à l'ensemble S_m^l son statut de solution. L'ensemble S_m^l ainsi défini existe et est unique ([BON 01]).

2.2. Algorithme d'adaptation

Nous décrivons dans cette section le principe général d'un algorithme permettant de calculer le sous-ensemble de demandes maximisant l'expression (3). Cet algorithme utilise la technique de programmation dynamique exposée dans [KOO 77].

2.2.1. Programmation dynamique discrète

Dans le cas discret, la programmation dynamique permet d'étudier un processus de décision fini en plusieurs étapes, chaque étape impliquant un choix parmi un nombre fini de possibilités. A chaque étape est associé un ensemble d'états et à chacun de ces états un ensemble de transitions vers des états de l'étape suivante. Un état correspond à une des décisions possibles à une étape donnée. Les transitions associées définissent quant à elles les états accessibles à l'étape $i + 1$ à partir d'un état donné à l'étape i (c'est-à-dire les décisions possibles à l'étape $i + 1$ si l'on opte pour la décision associée à l'état considéré de l'étape i). Chaque transition se voit en outre associer un coût. La programmation dynamique permet de déterminer une séquence de décisions conduisant à une somme des coûts des transitions correspondantes qui soit maximale. La première et la dernière étape comportent un unique état, desquels sont issues (ou aboutissent) un ensemble de transitions permettant de déterminer le coût associé respectivement à la première et à la dernière décision. Ce processus de décision peut être représenté sous forme d'un graphe, comme illustré à la figure 1. Nous y désignons par X^i l'ensemble des états de l'étape i et par x_j^i un élément quelconque de cet ensemble, $j = 1, \dots, |X^i|$.

Le chemin optimal est déterminé en progressant de la dernière étape vers la première, ce qui correspond à un parcours du graphe de la droite vers la gauche. Pour l'étape i et pour chaque état x_j^i au sein de cette étape, la séquence optimale ayant pour état initial x_j^i est calculée sur base des séquences optimales auparavant calculées pour l'étape $i + 1$ et des coûts associés aux transitions issues de x_j^i . Le processus se termine avec un chemin optimal issu de l'état initial.

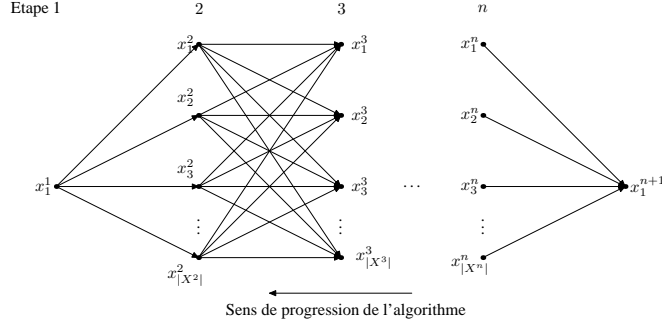


Figure 1. Graphe associé au processus de décision

La détermination du chemin optimal issu d'un état donné x_j^i à l'étape i s'effectue de la manière suivante. Nous désignons la séquence d'états optimale issue d'un état quelconque x par $\mathcal{P}(x)$, l'ensemble des états accessibles à partir de x par $\sigma(x)$, le coût associé à la transition entre deux états x et y par $C(x, y)$ et le coût associé à un chemin optimal $\mathcal{P}(x)$ par $C(\mathcal{P}(x))$. Pour chaque élément s de $\sigma(x_j^i)$, le coût $C(x_j^i, s)$ de la transition entre x_j^i et s est ajouté au coût total de $\mathcal{P}(s)$, le chemin optimal $\mathcal{P}(s)$ issu de s ayant été calculé précédemment. Si \bar{s} désigne l'élément de $\sigma(x_j^i)$ donnant lieu au coût résultant maximum, alors $\mathcal{P}(x_j^i)$ s'obtient simplement en ajoutant x_j^i en tête du chemin optimal issu de \bar{s} , $\mathcal{P}(\bar{s})$. Il s'agit d'une simple application du principe d'optimalité de Bellman : si l'on considère un état quelconque de la séquence de décision optimale, la séquence résultante issue de cet état est également optimale.

Considérons un processus de décision en n étapes. D'un point de vue algorithmique, nous effectuons une série de n itérations² partant de la droite du graphe, la $n - i + 1$ ^e itération, i allant de n à 1, ayant pour invariant que $\mathcal{P}(x_j^i)$ est connu pour tout $x_j^i \in X^i$. La validation initiale de l'invariant se réalise aisément à partir de l'état puits x_1^{n+1} : chaque état de l'étape n possède une unique transition vers cet état, transition définissant le seul chemin possible à partir d'un état de l'étape n . Ce chemin est donc nécessairement optimal. L'invariant est en outre rétabli au terme de chaque itération. En effet, lors de la $n - i + 1$ ^e itération, l'on recherche pour chaque état x_j^i l'état x_k^{i+1} vérifiant

$$\max_{x_k^{i+1} \in X^{i+1}} C(x_j^i, x_k^{i+1}) + C(\mathcal{P}(x_k^{i+1}))$$

$\mathcal{P}(x_k^{i+1})$ ayant été déterminé lors de l'itération précédente. $\mathcal{P}(x_j^i)$ est alors obtenu en ajoutant x_j^i à $\mathcal{P}(x_k^{i+1})$. Le principe d'optimalité de Bellman garantit que le chemin

2. Nous attirons l'attention du lecteur sur la distinction entre une *étape* du processus de décision et une *itération* de l'algorithme : une succession d'étapes correspond à un parcours du graphe dans le sens gauche-droite, tandis qu'une succession d'itérations correspond à un parcours dans le sens inverse.

ainsi construit est bien optimal. Finalement, la dernière itération calcule $\mathcal{P}(x_1^1)$, la solution du problème posé.

2.2.2. Application au problème étudié

Le problème de maximisation (3) peut être envisagé sous la forme d'un processus de décision impliquant n_{\max} étapes telles que l'étape i consiste en le choix du débit pour la couche i . Les couches sont envisagées dans l'ordre à partir de la couche de base, ce qui signifie que le débit choisi à l'étape $i + 1$ appartiendra nécessairement à l'ensemble des demandes supérieures au débit retenu pour l'étape i .

L'étape 1 comporte un seul état associé à la couche de base : celle-ci étant systématiquement sélectionnée, il n'y a en effet qu'un seul choix possible à cette étape. L'étape 2 comprendra quant à elle autant d'états qu'il y a de débits possibles pour la couche 2 parmi l'ensemble de demandes D et ainsi de suite. Le coût associé à une transition entre deux états x_j^i et x_k^{i+1} est égal au niveau de satisfaction obtenu par l'ensemble des récepteurs associés aux demandes comprises entre celles associées aux deux états considérés lorsqu'ils se rabattent sur x_j^i :

$$C(x_j^i, x_k^{i+1}) = \sum_{\substack{D_k \in D \\ D(x_j^i) \leq D_k < D(x_k^{i+1})}} w_k \frac{D(x_j^i)}{D_k} \quad (5)$$

où $D(x)$ désigne la demande de débit associée à l'état x . Un état puits final $x_1^{n_{\max}+1}$ est également introduit. Une unique transition relie chaque état de l'étape n_{\max} à cet état puits. Le coût associé est

$$C(x_i^{n_{\max}}, x_1^{n_{\max}+1}) = \sum_{\substack{D_k \in D \\ D(x_i^{n_{\max}}) \leq D_k}} w_k \frac{D(x_i^{n_{\max}})}{D_k} \quad (6)$$

Dans les relations (5) et (6), nous avons respectivement $D(x_j^i) = \bar{D}_k$ et $D(x_i^{n_{\max}}) = \bar{D}_k$; la somme des coûts C^i d'un chemin dans le graphe reliant les états x_1^1 et $x_1^{n_{\max}+1}$ est donc

$$\sum_{i=1}^{n_{\max}} C^i = \sum_{D_k \in D} w_k \frac{\bar{D}_k}{D_k} \quad (7)$$

A l'issue de l'algorithme décrit, une séquence de n_{\max} demandes a été sélectionnée dans D de manière à maximiser la somme (7), qui n'est autre que l'expression (3). Nous obtenons ainsi la solution du problème d'optimisation discret formulé précédemment. La condition (4) est quant à elle satisfaite en ne retenant à chaque itération que le chemin optimum débutant par la demande la plus faible, dans le cas où plusieurs chemins de même coût maximum seraient issus d'un état donné.

Toutes les demandes ne sont pas accessibles à toutes les étapes et toutes les successions de demandes ne sont pas non plus possibles. Cela se traduit de la manière suivante au niveau du graphe associé au processus de décision. Les états d'une étape

i donnée sont associés à un intervalle de demandes contiguës dont la borne inférieure est D_i et la borne supérieure $D_{m-(n_{\max}-i)}$. En effet, chaque étape implique la sélection d'une demande supérieure aux demandes déjà sélectionnées. Il n'est donc pas possible de sélectionner une demande inférieure à D_i à l'étape i . Cela se traduit également au niveau des transitions : il n'existe pas de transitions d'un état vers un autre pour lequel la demande associée serait inférieure ou égale. Il faut aussi veiller à ce qu'il reste suffisamment de demandes disponibles pour sélectionner n_{\max} débits au terme du processus. La plus grande demande que l'on puisse sélectionner à l'étape i est donc $D_{m-(n_{\max}-i)}$.

La détermination du sous-ensemble de demandes optimal se fait en temps polynomial. En effet, le calcul des coûts peut être réalisé avec un nombre constant d'opérations par coût. Le nombre de coûts étant de $\frac{m+2}{2}(m-1)+1$, cette étape possède une complexité $\mathcal{O}(m^2)$. La sélection du chemin s'effectue ensuite en n_{\max} itérations. Chacune de ces itérations implique l'examen de l'ensemble des états de l'étape associée, ensemble dont la taille est bornée strictement par m . Ensuite, pour chaque état, l'examen des transitions vers ses successeurs nécessite un nombre constant d'opérations. Chaque itération possède donc une complexité $\mathcal{O}(m^2)$, ce qui confère à l'algorithme une complexité globale $\mathcal{O}(n_{\max}m^2)$. Notons finalement que l'algorithme comprend une phase préalable de tri des demandes reçues en vue de générer l'ensemble D . Cette étape n'augmente toutefois pas la complexité de l'algorithme.

3. Mise en œuvre de l'adaptation dynamique

3.1. Calcul des demandes de débits

La mise en œuvre de l'algorithme d'adaptation présenté ci-dessus requiert le calcul par les récepteurs de demandes de débits et la transmission de ces dernières à la source. Au sein de l'architecture présentée, le calcul des demandes est un problème orthogonal à celui de l'adaptation. Les récepteurs peuvent en effet calculer leurs demandes selon un schéma quelconque. La seule contrainte de cohérence imposée par le problème d'optimisation (3) se situe au niveau de la mise à jour par les récepteurs de leur niveau d'abonnement : nous avons en effet supposé qu'un récepteur sélectionnait dans l'offre de la source le plus haut niveau d'abonnement qui soit inférieur ou égal à sa demande.

Dans le cas où les récepteurs mettent en œuvre un mécanisme de contrôle de congestion, la détermination des demandes de débits peut naturellement s'intégrer à l'action spécifique des récepteurs. A titre d'illustration, nous envisageons le cas d'un mécanisme de contrôle de congestion en boucle fermée visant l'équité vis-à-vis des flux TCP. L'objectif d'équité est atteint par les récepteurs via un calcul de débit admissible basé sur une équation provenant d'un modèle analytique de TCP. L'équation utilisée découle d'un modèle stochastique. Elle a été proposée dans [PAD 98] et reprise dans le cadre du protocole TFRC présenté dans [FLO 00] :

$$T = \frac{MSS}{RTT \sqrt{\frac{2p}{3}} + t_{RTO} \min \left(1, 3 \sqrt{\frac{3p}{8}} \right) p (1 + 32p^2)} \quad (8)$$

où T est le débit effectif de la connexion, MSS est la taille maximale des paquets transmis (*Maximum Segment Size*), RTT est le Round Trip Time, p le taux de pertes de paquets et t_{RTO} est la valeur du timer de retransmission TCP.

La valeur du timer de retransmission étant uniquement fonction du RTT, l'évaluation de la formule (8) ne requiert de la part des récepteurs que la connaissance du RTT et du taux de pertes. Si nous supposons ces deux grandeurs connues, chaque récepteur peut en déduire via l'équation (8) la valeur du débit d'un flux TCP placé dans les mêmes conditions. Le contrôle de congestion consiste alors pour le récepteur à sélectionner le plus haut niveau d'abonnement qui soit inférieur ou égal au débit calculé, le choix de la formule utilisée offrant des garanties d'équité vis-à-vis des flux TCP. La demande de débit est quant à elle la valeur résultant de l'évaluation de l'équation (8).

Un compromis doit être trouvé entre la précision de la mesure par les récepteurs de leur RTT et la résistance à un facteur d'échelle de la méthode utilisée. Chaque récepteur évalue exactement son RTT lors de son adhésion au groupe via l'envoi d'un paquet à la source. La source plaçant une estampille temporelle dans chaque paquet, le récepteur connaît les intervalles de temps entre paquets à la source. En mesurant ces mêmes intervalles lors de la réception, il peut en déduire les variations du RTT initialement mesuré. La valeur calculée dérive cependant progressivement de la valeur réelle. Une resynchronisation périodique est donc nécessaire ; celle-ci a lieu lors de l'envoi par le récepteur de sa demande de débit à la source.

Le calcul du taux pertes est un problème complexe. Nous utilisons une approche inspirée de [FLO 00] : le taux de pertes est calculé comme étant l'inverse d'une moyenne pondérée des intervalles de perte les plus récents, un intervalle désignant le nombre de paquets reçus entre deux pertes consécutives. Ce type de méthode présente toutefois certains défauts intrinsèques pour lesquels nous avons prévu des corrections. Une discussion détaillée de la méthode complète ainsi que du mécanisme de calcul du RTT peut être trouvée dans [BON 01].

3.2. *Transmission des demandes et résistance à un facteur d'échelle*

L'adaptation des débits est réalisée périodiquement par la source. Celle-ci utilise un marquage du flux de données pour notifier aux récepteurs d'envoyer leurs demandes. Les résultats de la dernière adaptation sont en outre régulièrement émis par la source via des paquets spécifiques. Les détails des opérations réalisées au cours de chaque période sont repris dans [BON 01].

La croissance de la taille du groupe multipoint est susceptible de mettre à mal l'architecture proposée. L'envoi d'un nombre élevé de demandes par les récepteurs risque de surcharger à la fois le réseau et la source, cette dernière disposant d'une capacité limitée en termes de réception et de traitement des demandes au travers de l'algorithme d'adaptation.

Nous proposons un mécanisme probabiliste de limitation du nombre R de demandes émises par les récepteurs. La source calcule la valeur d'une certaine probabi-

lité p , communiquée ensuite aux récepteurs. Les récepteurs envoient par la suite leur demande avec une probabilité p , ce qui statistiquement entraîne la réception par la source de $R \approx pG$ demandes, où G désigne le nombre de récepteurs. Connaissant p et R , la source peut estimer la taille du groupe. Il lui est donc possible au travers du choix de p de faire évoluer le nombre de demandes reçues en fonction de la taille réelle du groupe selon une courbe arbitrairement choisie. Idéalement, la valeur initiale de la probabilité devrait être fixée en fonction de l'application considérée et de la taille attendue du groupe lorsque la source se met à émettre.

Nous pouvons en toute généralité associer une méthode de calcul de la probabilité à chacune des entités devant être protégée d'une éventuelle surcharge. La méthode retenue lors d'une adaptation donnée est alors simplement la plus contraignante, pour laquelle la valeur calculée de la probabilité est la plus basse.

Nous utilisons un même schéma de calcul de la probabilité associée aux différentes entités. Chaque entité se voit associer un seuil en dessous duquel aucune limitation du nombre de demandes ne doit être envisagé — la probabilité est alors fixée à 1 — et une fonction de limitation traduisant pour cette entité l'évolution souhaitée du nombre de demandes en fonction de la taille du groupe. Lorsque le nombre R de demandes transmises dépasse le seuil, nous le faisons évoluer selon la fonction de limitation spécifique à l'entité et traduisons la valeur souhaitée de R en une valeur de p via la relation $R \approx pG$. Dans notre cas, nous avons opté pour une fonction de limitation logarithmique pour le réseau et constante pour la source, dont la capacité est connue a priori. Les détails relatifs aux fonctions choisies peuvent être trouvés dans [BON 01].

L'approche probabiliste présentée a été conçue dans l'objectif de conférer à l'architecture une résistance face à la croissance du *nombre* de demandes transmises. Nous n'avons pas considéré l'impact de cette limitation sur l'information disponible pour la source. La qualité de l'adaptation pourrait bien évidemment être remise en cause si la source ne recevait pas suffisamment de demandes pour couvrir l'hétérogénéité de ses récepteurs. Toutefois, les seuils des fonctions de limitation seront en pratique suffisamment élevés pour garantir la réception par la source d'un nombre de demandes couvrant les capacités de n'importe quel ensemble réaliste de récepteurs.

4. Evaluation de l'architecture proposée

Nous avons utilisé le simulateur de réseaux *ns* pour mener à bien une évaluation de l'architecture proposée. Les résultats présentés dans cette section se veulent représentatifs et non exhaustifs. Une analyse plus complète de l'architecture peut être trouvée dans [BON 01].

Nous utilisons la topologie de la figure 2. Nous y plaçons une source au nœud 0 et répartissons 19 récepteurs sur l'ensemble de la topologie. Nous envisageons tout d'abord un scénario où la source démarre en $t = 0.1s$ et l'ensemble des récepteurs rejoint celle-ci en $t = 0.2s$. La période d'adaptation résultant du choix des différents paramètres temporels du cycle d'adaptation est de $1.6s$.

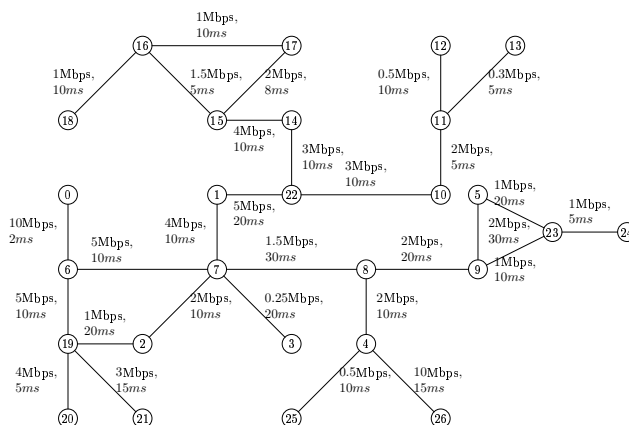


Figure 2. Topologie utilisée pour les simulations

L'objectif principal visé par l'architecture développée est que les récepteurs d'un groupe multipoint puissent utiliser au mieux les ressources dont ils disposent dans le cadre d'une transmission vidéo en couches. Ceci requiert d'une part que les récepteurs estiment correctement leur capacité et d'autre part que la source soit en mesure d'offrir un ensemble de débits adéquats.

Afin de juger si cet objectif est atteint, nous avons tout d'abord analysé l'évolution temporelle des débits cumulés des différents récepteurs dans un scénario où le flux en couches est le seul présent au sein de la topologie. Le nombre maximum n_{\max} de couches émises par la source est de 8, ce qui est un nombre suffisant pour couvrir de manière assez large l'ensemble des demandes. Les récepteurs prennent effectivement possession de la bande passante dont ils disposent et ce à un rythme inversement proportionnel à leur RTT, comme prévu par l'équation (8). Malgré l'offre réduite de la source par rapport à la taille du groupe, le niveau de satisfaction de chaque récepteur — mesuré par le rapport entre le débit associé au niveau d'abonnement et le débit demandé — est élevé, ce qui constitue un bon indicateur de l'adéquation de l'adaptation.

A titre d'illustration, nous présentons à la figure 3 l'évolution temporelle de l'offre de la source et du débit cumulé reçu par une série de nœuds représentatifs de l'hétérogénéité du groupe. Nous pouvons observer que les récepteurs associés à ces nœuds acquièrent relativement vite la bande passante dont ils disposent ; le débit reste ensuite stable. Dans ce scénario où nous laissons le système atteindre un état stationnaire, chaque récepteur se stabilise autour d'un niveau d'abonnement spécifique avec de temps à autres de brefs passages aux niveaux adjacents. Le récepteur situé au nœud 17 voit ainsi son débit cumulé subir une variation en forme d'échelon aux alentours de $t = 250s$. Il s'agit simplement du passage du niveau 5 au niveau 6 de l'abonnement de ce récepteur, passage consécutif à l'augmentation de sa demande de débit.

Nous pouvons aussi observer que, durant de brèves périodes, le débit de chaque couche diminue et prend pour valeur le débit de la couche immédiatement inférieure.

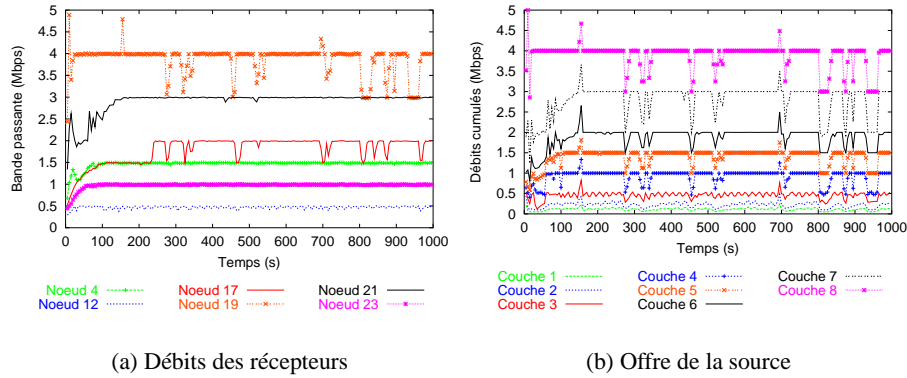


Figure 3. Evolution temporelle des débits des récepteurs et de l'offre de la source ($n_{max} = 8$)

Il s'agit de circonstances où, malgré l'état stationnaire, le nombre de demandes parvenant à la source diminue d'une unité suite à la réception d'une demande tardive arrivant après l'exécution de l'algorithme. La demande non comptabilisée étant un des éléments du sous-ensemble optimal recherché par l'algorithme, la répartition des couches s'en trouve modifiée. Le résultat de l'adaptation est dans notre scénario tel que le débit de la couche associée à la demande ignorée — en l'occurrence la couche 4 — est modifié et que les couches supérieures se voient décalées vers le bas. Les débits cumulés associés aux récepteurs suivent logiquement les variations dans l'offre de la source.

Lorsque l'ensemble des demandes ne peut être couvert aussi complètement — par exemple en choisissant la valeur 4 pour n_{max} —, nous observons que le niveau de satisfaction de certains récepteurs diminue logiquement, comme illustré à la figure 4.

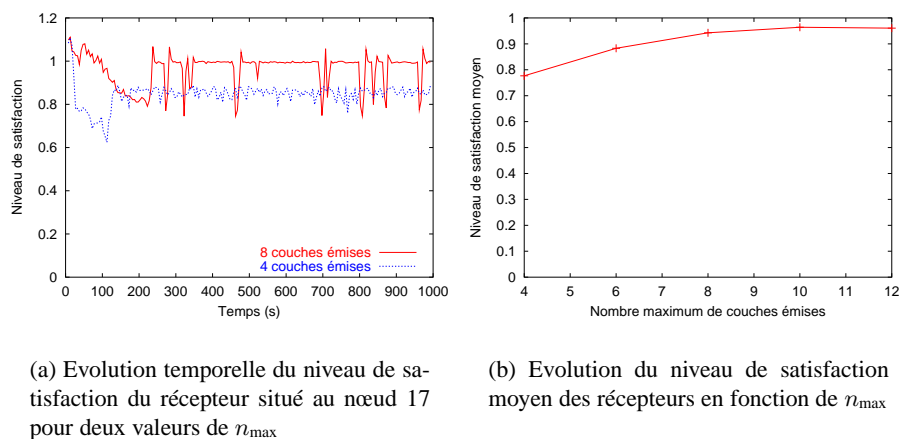


Figure 4. Influence sur les récepteurs de nombre de couches émises par la source

Nous y présentons également l'évolution du niveau de satisfaction moyen de l'ensemble des récepteurs en fonction du nombre de couches émises par la source. La courbe résultante, naturellement convexe, montre qu'un nombre de couches faible par rapport à la taille du groupe permet pour la topologie envisagée d'obtenir un niveau de satisfaction moyen élevé. Il faut toutefois tenir compte du fait que le niveau de satisfaction d'un récepteur peut ponctuellement être supérieur à l'unité en raison de l'existence d'un mécanisme d'hystérésis empêchant un récepteur de quitter un niveau dès que sa demande devient inférieure au débit associé. Le niveau de satisfaction moyen s'en trouve ainsi augmenté mais cet effet n'est pas suffisamment important pour remettre en cause les résultats obtenus.

Nous pouvons raisonnablement penser que la courbe obtenue ne dépend pas fondamentalement de la taille du groupe mais plutôt de son hétérogénéité. Les résultats obtenus peuvent donc s'étendre à des groupes de tailles différentes présentant une courbe de répartition des débits analogue.

Nous avons également étudié la manière dont l'offre de la source s'adaptait à des variations soudaines des capacités des récepteurs en introduisant dans la topologie des flux UDP à débits constants. Dans le cadre de leur contrôle de congestion, les récepteurs détectent rapidement le changement dans les conditions du réseau, que ce soit en terme de diminution de la bande passante disponible ou d'augmentation du RTT et du taux de pertes. Les récepteurs concernés diminuent en conséquence leur niveau d'abonnement et leur demande de débit. La mise à jour des demandes permet à la source de rapidement adapter son offre aux nouvelles attentes des récepteurs et ainsi de maintenir un niveau de satisfaction moyen élevé.

La figure 5 illustre l'évolution de la bande passante pour un récepteur situé au nœud 23 en présence d'un flux UDP dans le cas où n_{\max} vaut 8. Le flux UDP démarre en $t = 600s$ et s'établit entre les nœuds 7 et 9 avec un débit de 800Kbps. L'établissement de ce flux non adaptatif sur un chemin déjà saturé entraîne une brusque montée du taux de pertes pour les récepteurs en aval. Ceux-ci voient alors leurs demandes chuter avant de progressivement reprendre possession de la bande passante. Les récepteurs touchés sont globalement abonnés au niveau 4, niveau qui suit l'évolution à la baisse de la demande des récepteurs. La source adapte donc bien ses débits en fonction de l'évolution des capacités de ses récepteurs. Notons que le même genre de comportement serait observé dans le cadre d'une étude de l'impact de la dynamique du groupe sur la qualité de l'adaptation.

En ce qui concerne la résistance de l'architecture à un facteur d'échelle, l'évolution de la probabilité calculée par la source ainsi que du nombre de demandes lui parvenant a été confrontée aux courbes théoriques dérivant des fonctions de limitation utilisées. La correspondance entre ces courbes s'est révélée être très bonne ([BON 01]).

De même, nous avons étudié les qualités de l'architecture en terme d'équité vis-à-vis des flux TCP. Notre objectif n'était pas tant de montrer que notre architecture possédait de bonnes propriétés d'équité que d'insister sur le fait que de tels mécanismes pouvaient naturellement s'y intégrer. Les résultats obtenus en terme de répartition de la

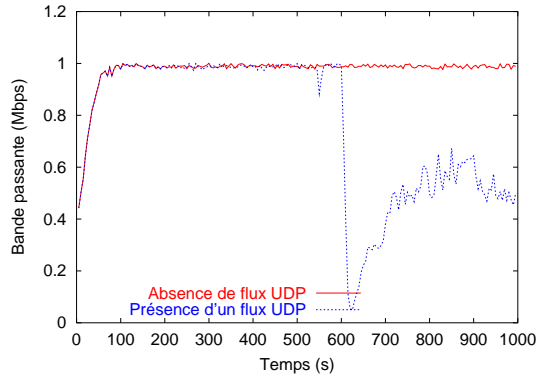


Figure 5. Evolution temporelle de la bande passante pour un récepteur situé au nœud 23

bande passante entre les deux types de flux ont largement confirmé les possibilités de mise en œuvre de politiques d'équité au sein de l'architecture développée ([BON 01]).

5. Conclusions

Nous avons proposé une architecture permettant la mise en œuvre d'une adaptation dynamique des débits des couches dans le cadre d'une transmission vidéo multipoint. Bien que l'accent ait été mis sur l'adaptation dynamique, nous nous sommes efforcés de construire une architecture complète et modulaire permettant de viser une série d'objectifs connexes comme le contrôle de flux, l'équité vis-à-vis des flux TCP ou encore la résistance à un facteur d'échelle. Il est à cet égard intéressant de remarquer que l'ensemble des mécanismes proposés peuvent être librement adaptés au sein de l'architecture. Ainsi, l'algorithme d'adaptation pourrait être redéfini en fonction d'une toute autre postcondition. Nous pourrions également fixer avant l'exécution de l'algorithme d'autres débits que celui de la couche de base, de manière à systématiquement offrir un ensemble représentatif de débits. Les récepteurs pourraient aussi de leur côté utiliser un autre schéma de contrôle de congestion et de calcul des demandes sans que les autres fonctionnalités soient remises en cause. Nous ne demandons par ailleurs aux nœuds intermédiaires aucun support autre que le routage multipoint.

L'évaluation de l'architecture a été réalisée via des simulations dont nous avons présenté ici un aperçu. Celles-ci ont permis de confirmer le bon fonctionnement des mécanismes développés sur un plan théorique de même que certaines limitations de ceux-ci. Nous souhaitons ainsi poursuivre le développement de l'architecture, notamment en vue de stabiliser davantage le système. Ceci requiert un lissage accru des demandes des récepteurs et de l'offre de la source. Cette dernière pourrait par exemple utiliser les poids apparaissant dans la formule (2) pour tenir compte dans une certaine mesure des demandes des adaptations précédentes. Il faut toutefois veiller à ne pas nuire à la dynamique de l'ensemble.

Il serait également intéressant d'étudier des méthodes de résistance à un facteur d'échelle plus sophistiquées, telles que la suppression de réponses présentée dans [NON 99] et reprise dans [SIS 00]. Soulignons à nouveau qu'un tel mécanisme pourrait s'intégrer directement dans notre architecture.

Nous projetons aussi de mener à bien des tests réels à l'aide d'un prototype en cours de développement. Ce dernier réalise une répartition entre les différentes couches des coefficients issus d'une transformée en cosinus discret d'une image capturée. Les couches basses reçoivent les coefficients associés aux basses fréquences et transportent donc l'information de base de l'image, le raffinement apporté par les hautes fréquences étant réservé aux couches supérieures. Le procédé génère actuellement un flux vidéo constitué d'une succession d'images fixes n'exploitant pas la redondance temporelle de ces dernières. Le codage JPEG200 actuellement en cours de standardisation devrait offrir une granularité plus fine pour les débits et le nombre de couches. Nous projetons à terme d'exploiter cette technique de codage.

6. Bibliographie

- [BON 01] BONMARIAGE N., « Conception et évaluation d'une méthode d'adaptation dynamique des débits des couches en transmission vidéo multipoint », rapport, 2001, Université de Liège, 125 pages, <http://www.run.montefiore.ulg.ac.be/~bonmarge>.
- [El 01] EL KHAYAT I., LEDUC G., « A stable and flexible TCP-friendly congestion control protocol for layered multicast transmission », *Proc. of 8th International Workshop IDMS'2001, LNCS 2158*, 2001, p. 154-167.
- [FLO 00] FLOYD S., PADHYE J., WIDMER J., « Equation-based Congestion Control for Unicast Applications », *ACM SIGCOMM*, vol. 30, n° 4, 2000.
- [KOO 77] KOO D., *Elements of Optimization*, Springer-Verlag, 1977.
- [LI 96] LI X., AMMAR M., « Bandwidth control for replicated-stream multicast video distribution », *Fifth IEEE International Symposium on High Performance Distributed Computing*, New York, August 1996.
- [MCC 96] MCCANNE S., JACOBSON V., VETTERLI M., « Receiver-driven Layered Multicast », *ACM SIGCOMM*, vol. 26,4, 1996.
- [NON 99] NONNENMACHER J., BIRSACK E. W., « Scalable Feedback for Large Groups », *IEEE/ACM Transactions on Networking*, vol. 7, n° 3, 1999, p. 375-386.
- [PAD 98] PADHYE J., FIROIU V., TOWSLEY D., KUROSE J., « Modeling TCP Throughput : A Simple Model and its Empirical Validation », *ACM SIGCOMM*, vol. 28, n° 4, 1998.
- [RUB 99] RUBENSTEIN D., KUROSE J., TOWSLEY D., « The Impact of Multicast Layering on Network Fairness », *Computer Communication Review*, vol. 29, n° 4, 1999.
- [SIS 00] SISALEM D., WOLISZ A., « MLDA : A TCP-friendly Congestion Control Framework for Heterogeneous Multicast Environments », *International Workshop on Quality of Service*, Pittsburgh, juin 2000.
- [VIC 98] VICISIANO L., RIZZO L., CROWCROFT J., « TCP-like congestion control for layered multicast data transfer », *IEEE INFOCOM*, mars 1998, p. 996-1003.
- [VIC 00] VICKERS B., ALBUQUERQUE C., SUDA T., « Source-Adaptative Multilayered Multicast Algorithms for Real-Time Video Distribution », *ACM Transactions on Networking*, vol. 8, n° 6, 2000.