

---

# Optimisation des débits des couches d'une transmission vidéo multipoint avec une meilleure prise en compte du surcoût d'encodage

Cyril Soldani — Guy Leduc

Université de Liège  
Research Unit in Networking  
Institut Montefiore (B28)  
B-4000 Liège (Sart-Tilman)  
{soldani,leduc}@run.montefiore.ulg.ac.be

---

*RÉSUMÉ.* Dans la transmission des flux vidéo, on propose souvent de résoudre le problème de l'hétérogénéité des récepteurs par la division des flux en couches cumulatives distribuées par transmission multipoint. Une adaptation dynamique par la source du nombre et du débit des couches à la population de ses récepteurs permet une meilleure utilisation du réseau. Des algorithmes efficaces ont été proposés pour réaliser cette adaptation. Néanmoins, l'encodage en couches d'un flux vidéo génère un surplus de débit par rapport à l'encodage du flux en une seule couche. Une modélisation trop naïve de ce surcoût entraîne une allocation de débit sous-optimale. Nous proposons une amélioration de l'algorithme d'adaptation du débit des couches par programmation dynamique pour lui permettre de tenir compte plus finement du surcoût et des contraintes de l'encodeur. À cette fin, nous introduisons un modèle de surcoût plus général. Comme notre algorithme est de complexité élevée, nous présentons également une heuristique permettant de trouver rapidement une solution proche de l'optimum et nous testons cette heuristique par simulations sur base de notre modèle.

*ABSTRACT.* In video stream transmission, one often proposes to solve the problem of the heterogeneity of the receivers by dividing the stream into several cumulative layers distributed over multicast transmission. A dynamic adaptation by the source of the number and size of layer bit-rates to the set of its receivers allows a better usage of the network resources. Efficient algorithms have been designed to do this adaptation. However, the encoding of a video stream in layers incurs an overhead relatively to the single-layer encoding of the same stream. A too naive modeling of this overhead causes a suboptimal allocation of bit-rate.

We propose an improvement of the layer bit-rate adaptation algorithm by dynamic programming to take a finer overhead model and other encoder constraints into account. To this end, we introduce a more general overhead model. As our algorithm is of high complexity, we also present a heuristic allowing to find quickly a solution which is near the optimum and we test this heuristic by simulations based on our model.

*MOTS-CLÉS :* vidéo, multipoint, adaptation dynamique

*KEY WORDS:* video, multicast, dynamic adaptation

---

## 1. Introduction

Un système de transmission multipoint d'un flux vidéo doit pouvoir gérer des différences importantes entre ses différents récepteurs (en terme de bande passante disponible, de résolution/fréquence supportée par le matériel, de préférence de l'utilisateur).

Trois approches sont utilisées pour résoudre ce problème :

**La transmission simultanée.** La vidéo est encodée plusieurs fois pour former des versions indépendantes, chaque fois avec un débit différent. Ce mécanisme est simple mais peu flexible et coûteux en ressources puisque la redondance entre les différentes versions n'est pas exploitée.

**L'encodage en couches cumulatives.** Le flux vidéo est décomposé en une *couche de base* et une ou plusieurs *couches de raffinement*. La couche de base peut être décodée indépendamment et fournit la qualité de base de la vidéo tandis que les couches de raffinement sont cumulatives (la  $i$ -ème couche nécessite également les couches 1 à  $i - 1$  pour pouvoir être décodée) et permettent d'améliorer progressivement la qualité de la vidéo décodée. Chaque couche est diffusée dans son propre groupe multipoint et les récepteurs peuvent adapter dans une certaine mesure le débit du flux reçu à leur capacité en s'abonnant à un nombre adéquat de couches. Cette méthode est plus complexe au niveau du codage mais elle est plus flexible et permet d'économiser les ressources du réseau en exploitant efficacement la redondance entre les différents niveaux de qualité offerts. C'est la méthode que nous exploitons dans cet article.

**L'encodage en descriptions multiples.** Cette méthode est similaire à la précédente à ceci près que les couches ne sont pas cumulatives. Chacune peut être décodée indépendamment mais la combinaison de plusieurs couches (appelées *descriptions*) permet d'affiner progressivement la qualité de la vidéo décodée. Cette méthode est encore plus flexible, mais l'exploitation moindre de la redondance entre les couches entraîne généralement une moindre économie de ressources que l'approche en couches cumulatives. Nous ne discuterons pas cette méthode plus avant.

Au début, les transmissions en couches ont été présentées comme des solutions plus efficaces que la transmission simultanée. Cependant, l'encodage en couche d'un flux vidéo demande plus de débit que l'encodage du même flux vidéo en une seule couche à qualité égale. Quand ce surplus de débit est élevé, la solution basée sur les transmissions simultanées peut se révéler plus efficace qu'une solution basée sur un encodage en couches dans certains scénarios (*e.g.* [KIM 05, AVR 07]). Avec l'amélioration des codeurs en couches cumulatives (principalement H.264/SVC [ALF 07, WIE 07]), il y a un regain d'intérêt pour cette approche.

Afin de faire une utilisation la plus judicieuse possible des ressources du réseau lors de la distribution en couches cumulatives (ou pour pouvoir la comparer/combiner avec l'approche basée sur la transmission simultanée), il faut une méthode pour adapter dynamiquement le nombre et le débits des couches à la population des récepteurs. Plusieurs algorithmes ont été proposés pour réaliser cette adaptation de manière efficace (*e.g.* [BON 02, FEI 03, KIM 05, YAN 00]).

Cependant, ces algorithmes ne prennent pas en compte le surplus de débit induit par l'encodage en couches (ou pas de manière suffisamment fine). Quand ce surcoût n'est pas pris en compte, l'avantage de l'encodage en (nombreuses) couches vis-à-vis d'autres approches comme la transmission simultanée de plusieurs flux est surévalué. D'autres contraintes liées à l'encodeur n'ont également été

que peu ou pas prises en compte et peuvent également induire une allocation des débits des couches sous-optimale.

Dans cet article, nous proposons des améliorations aux algorithmes existants, principalement sous la forme d'une prise en compte plus générique du surplus de débit lié à l'encodage en couches. Comme notre solution est de complexité élevée, nous présentons également une heuristique permettant de réduire la complexité tout en conservant une solution optimale ou proche de l'optimum.

L'article est organisé comme suit. La section 2 présente l'algorithme d'allocation des débits des couches par programmation dynamique qui nous a servi de base. Nous présentons ensuite dans la section 3 les améliorations proposées. La complexité de notre solution et les heuristiques applicables sont discutées à la section 4. Nous présentons une évaluation des améliorations proposées à la section 5. Enfin, nous concluons à la section 6.

## 2. Adaptation dynamique des débits des couches par programmation dynamique

### 2.1. Définition du problème

Nous supposons que les débits accessibles aux récepteurs sont principalement liés à leur matériel et à leur lien d'accès de telle sorte qu'ils soient indépendants les uns des autres. Nous appelons ces débits les *débits isolés* des récepteurs. Nous supposons qu'un mécanisme de retour d'information permet à la source de disposer de la liste de ces débits isolés. Plusieurs solutions ont été proposées pour estimer cette information d'une manière qui passe à l'échelle (e.g. [NON 99, BON 02]). Nous notons  $R = \{R_i\}, i = 1, \dots, m$  l'ensemble ordonné des débits isolés des  $m$  récepteurs.

Soit  $n \leq m$  le nombre de couches autorisées et  $C = \{C_i\}, i = 1, \dots, n$  l'ensemble ordonné des débits *cumulés* associés aux différentes couches, nous cherchons une relation permettant de déterminer  $C$  à partir de  $R$ .

Idéalement (*i.e.* en ignorant le surplus de débit lié à l'encodage et à la distribution en couche), chaque récepteur devrait pouvoir recevoir un flux dont le débit est égal à son débit isolé, soit  $n = m$  et  $C_i = R_i, i = 1, \dots, n$ .

Cependant, le nombre de couches est limité en pratique, pour diverses raisons :

- le nombre de groupes multipoints disponibles est limité ;
- l'utilisation d'un grand nombre de couches rend l'encodage et le décodage plus complexes ;
- l'utilisation d'un grand nombre de groupes augmente la part du trafic de signalisation réseau au détriment du trafic des données vidéo ;
- le découpage du flux vidéo en couches entraîne un surplus de débit, ce codage étant moins efficace qu'un codage mono-couche.

Il est donc raisonnable de supposer que le nombre de couches  $n$  est fixé *a priori* et est inférieur au nombre de récepteurs  $m$  (nous reviendrons sur le choix du nombre de couches à la section 3.3.2).

La source va essayer de maximiser la satisfaction de ses récepteurs. Différentes fonctions d'utilité peuvent être utilisées pour mesurer cette satisfaction. Nous avons retenu l'écart relatif moyen

entre les débits isolés des récepteurs et les débits auxquels ils pourront recevoir le flux vidéo sans congestion. Formellement, nous cherchons à minimiser

$$\sum_{i=1}^m \frac{R_i - \bar{R}_i}{R_i} \quad (1)$$

où  $\bar{R}_i$  est le plus grand débit effectivement offert par la source qui soit inférieur ou égal à  $R_i$  (*i.e.*  $\max(\{C_j \in C \mid C_j \leq R_i\})$ ). Nous pourrions prendre d'autres mesures d'utilité similaires.

Nous considérons dans un premier temps que le débit de la plus petite couche  $C_1$  doit correspondre au plus petit débit isolé  $R_1$  (nous voulons que tous les récepteurs puissent recevoir le flux). Le débit de la couche de base étant fixé, il reste à déterminer les débits des  $n - 1$  couches de raffinement de manière à minimiser (1) ce qui revient à maximiser

$$\mathcal{U}(C) = \sum_{i=1}^m \frac{\bar{R}_i}{R_i}. \quad (2)$$

Cette définition implique que le choix des débits offerts doit se faire parmi les débits isolés des récepteurs,  $C$  sera un sous-ensemble de  $R$ .

Nous aurions pu chercher à optimiser directement le PSNR ou le SSIM, mais ces métriques étant plus difficiles à modéliser ou à mesurer, nous nous sommes limités aux débits reçus.

## 2.2. Algorithme d'adaptation

Une solution efficace au problème de maximisation (2) repose sur la programmation dynamique discrète. La programmation dynamique permet d'analyser de manière simple et efficace un processus de décision fini.

L'algorithme procède par étapes. Chaque étape est associée à un ensemble d'états correspondant aux décisions possibles à cette étape. À chacun de ces états est associé un ensemble de transitions vers des états de l'étape suivante, indiquant les décisions possibles à l'étape suivante si on opte pour la décision associée à cet état à l'étape courante. Une utilité est associée à chaque transition. La programmation dynamique permet de trouver un chemin optimal entre les états tel que la somme des utilités de ses transitions soit maximale.

Dans le cas qui nous intéresse, les étapes correspondront aux couches, chaque état correspondant à un débit possible pour la couche. Les transitions représenteront les passages possibles du débit d'une couche  $i$  aux débits supérieurs de la couche  $i + 1$ . L'utilité de la transition sera la portion de la somme (2) qui concerne les récepteurs dont le débit isolé se situe dans  $[C_i, C_{i+1}[$ .

Le chemin optimal est déterminé en progressant à rebours de la dernière étape vers la première. On commence par déterminer l'utilité associée aux états (*i.e.* aux débits potentiels) de la dernière couche, en considérant que tous les récepteurs de débit isolé supérieur se rabattent sur cet état. Pour chaque état de la couche  $n - 1$ , on détermine ensuite le meilleur chemin ayant pour source cet état sur base de l'utilité de chacune de ses transitions et des utilités qui viennent d'être déterminées à l'étape

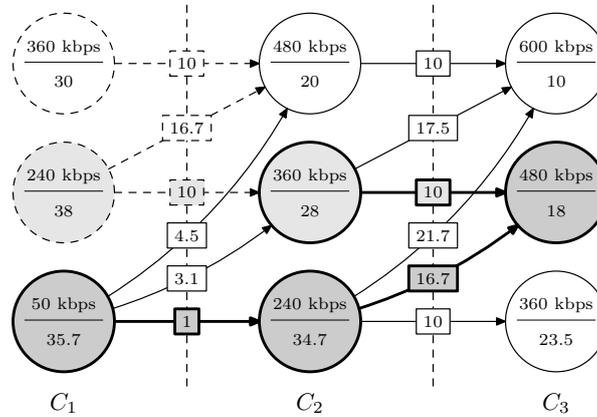


Figure 1 – Illustration de l'allocation de débit par programmation dynamique. Le nombre en dessous du débit représente l'utilité du meilleur chemin issu de cet état. Le chemin en gris foncé représente le meilleur chemin pour la couche de base à 50 kbps tandis que le chemin en gris clair donne le meilleur chemin si on se donne également la possibilité d'utiliser un des deux états en pointillés comme couche de base.

$n$ . On répète le processus jusqu'à atteindre l'état initial dans la couche 1. Le principe d'optimalité de BELLMAN garantit que le chemin ainsi obtenu est bien le chemin qui maximise la somme (2).

Cet algorithme a une complexité en  $\mathcal{O}(nm^2)$  et est donc plutôt efficace.

Un exemple concret est illustré à la figure 1. Il s'agit de l'allocation des débits de trois couches pour 41 récepteurs dont les débits isolés sont uniformément répartis dans l'ensemble  $\{240 \text{ kbps}, 360 \text{ kbps}, 480 \text{ kbps}, 600 \text{ kbps}\}$ , à l'exception d'un récepteur dont le débit isolé est de 50 kbps.

### 3. Améliorations

#### 3.1. Couche de base optimale

La couche de base a une importance particulière au niveau du codage. Choisir une couche de base petite peut avoir un impact négatif fort sur la qualité des couches supérieures et choisir la plus petite demande comme couche de base n'est pas nécessairement une bonne idée. Si nous laissons l'algorithme choisir la couche de base optimale et que celle-ci se révèle être plus grande que le plus petit débit isolé, nous avons là un indicateur fort en faveur d'une transmission simultanée d'une session pour les petits récepteurs et une pour les récepteurs plus capables.

Pour laisser l'algorithme déterminer la couche de base optimale, il suffit d'ajouter tous les autres états initiaux possibles au graphe. Le processus est également illustré à la figure 1.

Ce nouveau degré de liberté ne change pas la complexité de l'algorithme, presque tous les calculs étant partagés avec le cas précédent.

### 3.2. Optimisation conjointe

Adapter les paramètres de l'encodeur vidéo pour produire un flux vidéo découpé en couches dont les débits ont été choisis par l'algorithme d'allocation basé sur les débits isolés des récepteurs n'est pas toujours possible ou pratique. Même quand l'encodeur le permet, cette approche peut entraîner une baisse de l'efficacité de codage, qui peut à son tour entraîner une allocation sous-optimale (si la meilleure adaptation aux récepteurs est compensée par la perte d'efficacité à l'encodage).

Plutôt que de calculer l'allocation de débits optimale uniquement sur base des demandes des récepteurs (*i.e.* déterminer  $C$  comme sous-ensemble de  $R$ ), et puis d'essayer d'encoder la vidéo en fonction du résultat après coup, nous utilisons une optimisation conjointe pour l'allocation des débits des couches [YAN 00]. Nous cherchons toujours à minimiser les écarts relatifs entre les débits isolés des récepteurs et les débits qu'ils reçoivent, mais les débits qu'ils reçoivent sont maintenant choisis parmi un ensemble *raisonnable* pour l'encodeur et non plus parmi les débits demandés par les récepteurs.

Soit  $E = \{E_i\}$ ,  $i = 1, \dots, k$  l'ensemble des débits cumulés autorisés.  $E$  est défini de telle sorte que chaque  $E_i \in E$  corresponde à un encodage qui est à la fois faisable et raisonnablement efficace. Nous déterminons  $C$  comme un sous-ensemble de  $E$  plutôt que comme un sous-ensemble de  $R$ . La fonction d'utilité reste (2), mais elle est maintenant calculée partiellement sur  $E$  par notre nouvelle définition de  $C$ .

Pour continuer notre exemple, supposons que nous autorisons les débits suivants correspondant à différentes possibilités d'encodage de la séquence *football* avec H.264/SVC : {82 kbps, 211 kbps, 402 kbps, 507 kbps, 586 kbps}. Le processus d'optimisation conjointe donne le chemin optimal {211 kbps, 402 kbps, 586 kbps} avec une utilité de 32.8.

Nous observons que l'utilité est moindre que dans le cas idéal où l'encodeur peut produire efficacement tous les débits possibles. Par ailleurs, comme souvent dans les problèmes d'optimisation discrète, nous observons que la solution conjointe optimale est différente de la solution la plus proche de celle trouvée pour le cas idéal.

### 3.3. Prise en compte du surcoût

Plusieurs études (*e.g.* [KIM 05, AVR 07]) ont montré que le surplus de débit lié à l'encodage en couches ne pouvait pas être négligé. Les récepteurs ne sont pas intéressés par la maximisation du débit du flux vidéo reçu, mais bien par sa qualité. Négliger le surplus de débit induit par l'encodage en couches introduit un biais en faveur d'un plus grand nombre de couches pour maximiser les débits reçus, mais cela peut en pratique réduire la qualité du flux vidéo décodé. Nous devons donc prendre en compte ce surplus de débit pour trouver l'optimum réel, basé sur la qualité de la vidéo décodée par les récepteurs.

Nous proposons donc une modification de l'algorithme d'allocation pour prendre en compte ce surplus de débit. L'idée est de considérer le débit total d'une couche lorsque nous considérons les récepteurs qui peuvent la recevoir, mais de ne considérer que son *débit effectif* (*i.e.* le débit du flux vidéo mono-couche de même qualité) dans le calcul de l'utilité.

Nous maximisons toujours (2), mais  $\bar{R}_i$  représente maintenant le débit effectif de la plus haute couche dont le débit total est inférieur ou égal au débit isolé du récepteur  $i$ .

Nous avons donc besoin d'un modèle de surplus de débit pour pouvoir associer à chaque  $E_i \in E$  son débit effectif.

### 3.3.1. Modèle de surplus de débit

Le modèle le plus souvent utilisé pour modéliser le surplus de débit associé à l'encodage en couches consiste à considérer un surplus nul pour la couche de base (souvent encodée comme un flux mono-couche) et un surplus de débit proportionnel à leur débit ajouté pour chacune des couches de raffinement. Le surplus de débit proportionnel  $\epsilon$  varie généralement entre 10% et 30% (selon les codeurs et les séquences).

Ce modèle très simple prend en compte l'importance particulière de la couche de base mais n'est pas suffisamment réaliste pour les couches de raffinement. En effet, pour une couche de base donnée, il donnera le même débit effectif pour un débit total donné, et ce quel que soit le nombre de couches intermédiaires. En pratique, nous observons une dégradation des performances avec le nombre de couches. Pour prendre en compte cet aspect, nous ajoutons pour chaque couche de raffinement un surplus de débit fixe  $K$  représentant le fait que l'échange de paramètres liés à la gestion des couches ne peut pas descendre en dessous d'un certain seuil. Soit  $\mathcal{E}(C_i)$  le débit effectif de la couche  $i$ , nous avons :

$$\begin{cases} \mathcal{E}(C_0) = C_0 \\ \mathcal{E}(C_i) = \mathcal{E}(C_{i-1}) + \max((1 - \epsilon)(C_i - C_{i-1}) - K, 0) \end{cases} \quad (3)$$

Ce modèle n'est toujours pas satisfaisant, car le débit effectif apporté par plusieurs couches de raffinement est indépendant de leur ordre (*i.e.* ajouter une large couche puis une petite donne le même résultat qu'ajouter la petite, puis la grosse), ce qui ne se vérifie pas en pratique. Cependant, nous nous en contenterons pour illustrer notre approche. Un modèle de surcoût d'encodage réaliste peut en effet être assez complexe et dépendre de nombreux paramètres : nombre de couches, type d'extension (spatiale, temporelle, qualité), débits totaux et effectifs des couches précédentes, ...

[KIM 05] propose un modèle linéaire pour deux couches dans une gamme de bande passante donnée, obtenu sur base de données expérimentales. Nous avons également effectué nos simulations avec une variante de ce modèle, avec des résultats assez similaires à ceux qui sont présentés dans la suite pour notre modèle plus simple. Nous ne les avons donc pas reportés ici. [HSU 07] présente également un modèle expérimental pour l'impact du choix de la couche de base dans le cas de la décomposition fine (FGS, *fine grained scalability*).

Quel que soit le modèle choisi, pour autant qu'il soit suffisamment réaliste et général, le débit effectif d'une couche va dépendre des débits effectifs des couches précédentes, et donc des débits qui ont été choisis pour les couches précédentes. Ce couplage nous empêche de continuer à utiliser l'algorithme d'allocation par programmation dynamique à proprement parler, puisque nous ne pouvons plus considérer un état indépendamment du chemin qui y a mené. Nous reviendrons sur la complexité de ce changement à la section 4.

À la figure 2, nous reprenons notre exemple en tenant compte du surplus de débit estimé à l'aide de notre modèle (3).

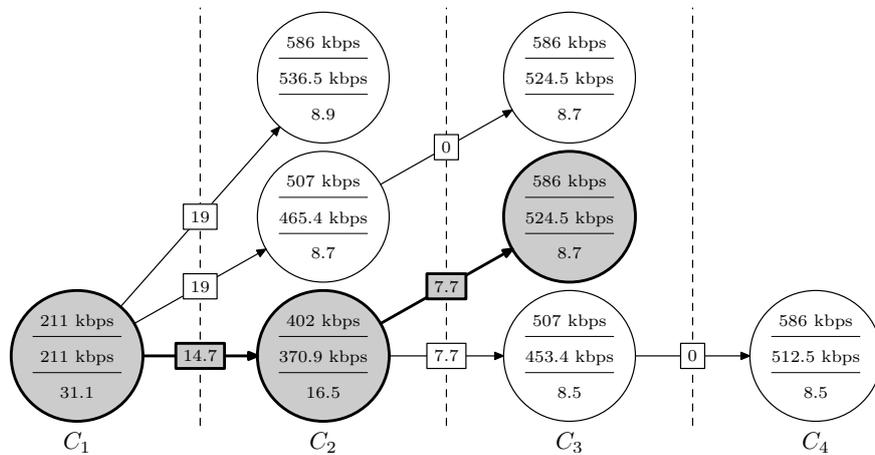


Figure 2 – Exemple d’optimisation prenant en compte le surplus de débit dû au codage en couches. Le second débit de chaque état représente l’estimation de son débit effectif. Seule la couche de base optimale est représentée.

### 3.3.2. Adaptation du nombre de couches

Quand nous ignorons le surplus de débit lié à l’encodage en couches, le nombre de couches optimal était toujours le nombre de débits d’encodage possibles  $k$  avec la solution triviale  $n = k$  et  $L = E$ . Le nombre de couches  $n$  était donné comme contrainte externe.

Maintenant que nous prenons le surcoût de codage et de transmission en compte, a meilleure solution n’est plus nécessairement celle qui comprend le plus grand nombre de couches et la question se pose de savoir quel est le nombre de couches optimal.

Plutôt que d’exécuter notre algorithme plusieurs fois avec divers nombres de couches autorisées, nous pouvons intégrer la sélection du nombre de couches directement dans le processus d’optimisation, ce qui nous permet de partager la plupart des calculs. Pour ce faire, il suffit de considérer le nombre maximum de couches autorisées et de permettre des transitions vers tous les débits supérieurs à toutes les étapes. Le nombre de couches optimal sera donné par le nombre d’états dans le chemin optimal.

## 4. Complexité

Lorsque l’on considère un modèle de surcoût dont le débit effectif d’une couche dépend des couches précédentes, chaque état de l’algorithme d’allocation par programmation dynamique est dupliqué autant de fois qu’il y a de chemins qui y mènent. En effet, chaque chemin donnera potentiellement un débit effectif différent et donc des utilités de transition différentes. Cette explosion du nombre d’états est exponentielle. Nous passons d’un algorithme de partitionnement (par programmation dynamique) à un algorithme d’exploration systématique de tous les sous-ensembles d’un ensemble. Nous nous attendons donc à une complexité exponentielle  $\mathcal{O}(2^k)$ .

Nous pouvons montrer que le nombre de calcul d'utilité nécessaires est donné par

$$2 \left( \sum_{i=0}^{n-1} \binom{k-1}{i} \right) + \left( \sum_{i=2}^n \binom{k-1}{i} \right) - 1 \quad (4)$$

ce qui donne  $3 \times 2^{k-1} - k - 1$  quand  $n = k$  et le nombre d'opérations est donc bien exponentiel en fonction du nombre de débits d'encodage possibles si on ne limite pas le nombre de couches autorisées.

Néanmoins, la situation n'est pas tout à fait aussi désespérée qu'il n'y paraît au premier abord. Pour tout  $n$  fixé, le nombre de calculs d'utilité est en effet donné par un polynôme de degré  $n$  en  $k$  et est donc de complexité asymptotique  $\mathcal{O}(k^n)$ . Nous reportons quelques nombres de calculs d'utilité en fonction du nombre de couches autorisées  $n$  et du nombre de débits d'encodage possibles  $k$  dans le tableau 1. Pour des valeurs raisonnables de  $n$  et surtout de  $p$ , l'algorithme reste utilisable en pratique.

$k/n$	3	4	5	6	7	...	20
3	8						
4	17	19					
5	31	40	42				
6	51	76	87	89			
7	78	133	169	182	184		
⋮	⋮	⋮	⋮	⋮	⋮	⋮	
20	1 521	7 335	26 715	77 103	181 755	...	1 572 843
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
32	5 953	46 408	279 249	1 355 352	5 457 489	...	6 045 186 714

Tableau 1 – Nombre de calculs d'utilité en fonction du nombre  $k$  de débits d'encodage possibles (lignes) et du nombre  $n$  de couches autorisées (colonnes).

Lors de l'implémentation, il n'est pas nécessaire de retenir tout le graphe en mémoire mais uniquement la branche courante (parcours en profondeur d'abord). L'algorithme consomme donc beaucoup de temps CPU mais presque pas de mémoire. Il se prête également très bien à la parallélisation.

## 5. Évaluation

Pour tester notre modèle et notre algorithme, nous avons choisi une gamme de débit d'encodage possibles compris entre 80 kbps et 1.1 Mbps (variation entre QCIF à 7.5 Hz et CIF à 30 Hz pour la séquence *football*). Nous avons fait varier le nombre de débits d'encodage possibles  $k$  entre 2 et 40. Les  $k$  débits d'encodage possibles sont répartis de telle manière que les débits ajoutés par les couches potentielles successives suivent une progression géométrique, ce qui est un modèle couramment utilisé. Les récepteurs, au nombre de 100, ont quant à eux été groupés en 20 classes de récepteurs de débits isolés  $\{80 \text{ kbps}, \dots, 1.1 \text{ Mbps}\}$ , selon une distribution binormale (avec  $\mu_1 = 350 \text{ kbps}$ ,

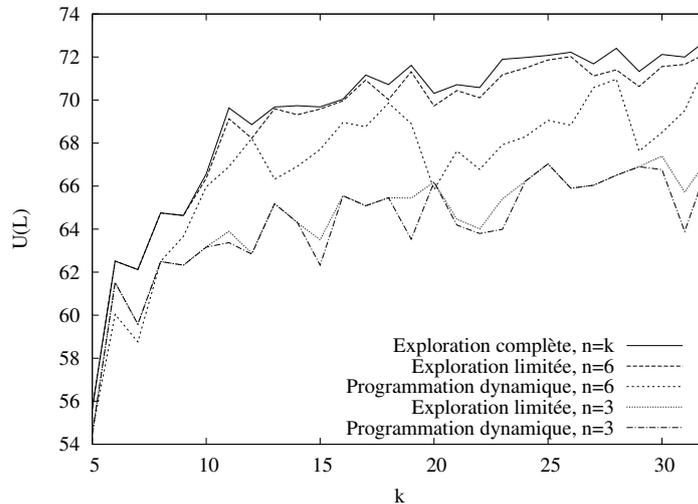


Figure 3 – Comparaison entre différents nombres de couches autorisées, avec ou sans prise en compte du surplus de débit.

$\sigma_1 = 3$ ,  $\mu_2 = 944$  kbps et  $\sigma_2 = 2$ ). Enfin, notre modèle a été paramétré avec  $\epsilon = 15\%$ , ce qui correspond au milieu de la gamme communément admise, et  $K = 15$  kbps.

La figure 3 montre l'évolution de l'utilité globale avec l'augmentation du nombre de débits d'encodage  $k$ . Nous avons représenté d'une part le résultat de l'exploration complète avec  $n = k$  et d'autre part les résultats des explorations limitées respectivement à 6 et 3 couches autorisées. Nous avons également représenté l'estimation de l'utilité fournie par notre modèle (3) pour les solutions obtenues par la méthode de programmation dynamique (*i.e.* sans tenir compte du surplus de débit).

Les courbes ne sont pas monotones. Cela est dû au fait que les débits d'encodage possibles se déplacent avec  $k$  dans notre simulation, rendant l'adaptation aux demandes des récepteurs plus ou moins facile. Nous nous approchons fortement de l'optimum avec la méthode par exploration limitée avec  $n = 6$  (l'optimum se situe souvent autour des 9 couches au maximum dans ces exemples). Cela nous conforte dans l'idée que nous pouvons réduire l'exploration à un faible nombre de couches sans trop nous éloigner de l'optimum. Enfin, si la méthode classique par programmation dynamique donne des résultats proches de l'exploration limitée pour les petits nombres de couches, nous observons une différence plus importante pour les plus hautes valeurs de  $n$ .

La figure 4 représente quant à elle l'utilité globale en fonction du nombre de couches autorisées  $n$  pour  $k = 32$ . Nous avons également représenté sur cette figure l'utilité *idéale*, *i.e.* celle qui serait obtenue avec  $n$  couches en l'absence de surplus de débit lié à l'encodage en couches. Cette utilité idéale est celle sur laquelle se base l'algorithme par programmation dynamique.

Le nombre de couches nécessaires pour atteindre l'optimum dans cette configuration est à nouveau faible. Même si la méthode par programmation dynamique donne de bons résultats pour  $n = 5$  ou 6, la méthode par exploration limitée donne tout de même une amélioration sensible. Par exemple,

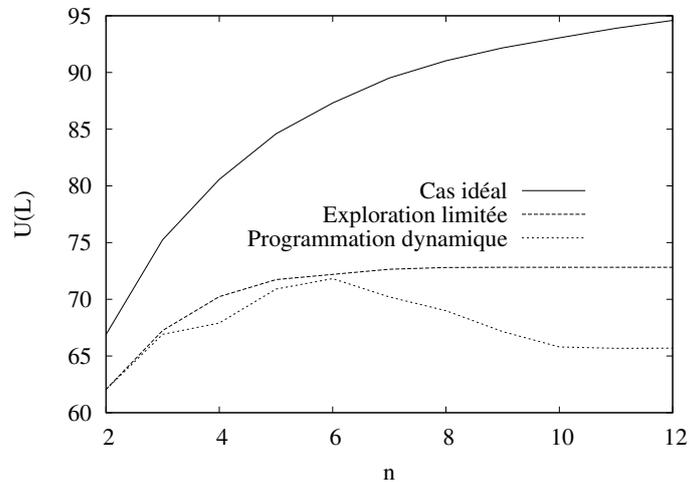


Figure 4 – Impact de la prise en compte du surplus de débit pour  $k = 32$ . La courbe intitulée *cas idéal* correspond à l'utilité qui serait obtenue s'il n'y avait aucun surplus de débit associé à l'encodage en couches.

la solution à 5 couches pour la méthode par exploration limitée donne une meilleure utilité que la solution à 6 couches (la meilleure) pour la méthode par programmation dynamique, ce qui permet d'économiser une couche tout en gagnant en qualité. Enfin, la méthode par programmation dynamique donne de mauvais résultats quand nous autorisons trop de couches. Cela est dû au fait que cette méthode ignore le surplus de débit et choisit toujours le nombre maximum de couches autorisées (entraînant une baisse de l'utilité quand la couche ajoutée entraîne plus de perte en surplus de débit que de gain en adaptation aux débits isolés des récepteurs). La méthode par exploration limitée va quant à elle sélectionner un nombre de couches inférieur à celui qui est autorisé dans ce cas (c'est pourquoi la courbe sature à partir de 8 couches autorisées).

## 6. Conclusions

Nous avons proposé une méthode d'allocation des débits des couches dans une transmission vidéo multipoint, qui permet de prendre en compte le surplus de débit lié à l'encodage en couches. Nous avons également présenté un modèle simple pour estimer ce surplus de débit.

Nous avons montré que malgré une complexité *a priori* exponentielle, la prise en compte du surplus de débit était réalisable en pratique, pour autant qu'on limite le nombre de couches autorisées. Nous avons montré qu'il était possible d'obtenir des solutions optimales ou proches de l'optimum dans des cas pratiques malgré cette restriction.

Le modèle simple que nous avons introduit ne fait pas ressortir une différence très importante entre l'optimisation des débits des couches par programmation dynamique ou par exploration limi-

tée, mais on observe tout de même une amélioration notable qui permet parfois de réduire le nombre de couches nécessaires à qualité égale.

En outre, même si la méthode par programmation dynamique peut donner de bons résultats pour de petits nombres de couches, nous avons montré qu'elle mérite une estimation de l'erreur due au surplus de débit *a posteriori* afin de bien sélectionner le nombre de couches. On pourrait en effet facilement le choisir trop petit (surévaluation de l'utilité globale atteinte) ou trop grand (l'effet du surplus de débit prenant alors le pas sur l'amélioration de l'adaptation aux récepteurs).

Cependant, il nous manque encore un bon modèle de surcoût et une évaluation expérimentale de celui-ci pour avoir une bonne base de comparaison entre les différentes méthodes. Nous comptons aussi développer d'autres heuristiques que la limitation du nombre de couches pour pouvoir traiter de plus grandes gammes de bande passante. Enfin, il serait intéressant d'optimiser conjointement la méthode par transmission en couches et la méthode par transmission simultanée.

## 7. Bibliographie

- [ALF 07] ALFONSO D., GHERARDI M., VITALI A., ROVATI F., « Performance Analysis of the Scalable Video Coding Standard », *Proc. of Packet Video*, Nov 2007, p. 243–252.
- [AVR 07] AVRAMOVA Z., VLEESCHAUWER D. D., SPAEY K., WITTEVRONGEL S., BRUNEEL H., BLONDIA C., « Comparison of Simulcast and Scalable Video Coding in Terms of the Required Capacity in an IPTV Network », *Proc. of Packet Video*, Nov 2007, p. 113–122.
- [BON 02] BONMARIAGE N., LEDUC G., « Adaptation dynamique des débits des couches pour la transmission vidéo multipoint », OBAID A., Ed., *Actes de Colloque Francophone sur l'Ingénierie des Protocoles (CFIP)*, Hermès Lavoisier, May 2002, p. 19–34.
- [FEI 02] FEI Z., AMMAR M. H., ZEGURA E. W., « Multicast Server Selection : Problems, Complexity, and Solutions », *IEEE Journal on Selected Areas in Communications*, vol. 20, n° 7, 2002, p. 1399–1413.
- [FEI 03] FEI Z., YANG M., AMMAR M. H., ZEGURA E. W., « A Framework for Allocating Clients to Rate-Constrained Multicast Servers », *Computer Communications*, vol. 26, n° 12, 2003, p. 1255–1262.
- [HSU 07] HSU C.-H., HEFEEDA M., FRASER S., « Optimal Partitioning of Fine-Grained Scalable Video Streams », *International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV)*, June 2007.
- [KIM 05] KIM T., AMMAR M. H., « A Comparison of Heterogeneous Video Multicast Schemes : Layered Encoding or Stream Replication », *IEEE Transactions on Multimedia*, vol. 7, n° 6, 2005, p. 1123–1130.
- [LIU 03] LIU J., LI B., ZHANG Y.-Q., « Adaptive Video Multicast over the Internet », *IEEE Multimedia*, vol. 10, n° 1, 2003, p. 22–33.
- [LIU 06] LIU J., LI B., ZHANG Y.-Q., « Optimal Stream Replication for Video Simulcasting », *IEEE Transactions on Multimedia*, vol. 8, n° 1, 2006, p. 162–169.
- [NON 99] NONNENMACHER J., BIRSACK E. W., « Scalable feedback for large groups », *IEEE/ACM Trans. Netw.*, vol. 7, n° 3, 1999, p. 375–386, IEEE Press.
- [WIE 07] WIEN M., SCHWARZ H., OELBAUM T., « Performance Analysis of SVC », *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, n° 9, 2007, p. 1194–1203.
- [YAN 00] YANG Y. R., KIM M. S., LAM S. S., « Optimal Partitioning of Multicast Receivers », *Proc. of International Conference on Network Protocols*, Nov 2000, p. 129–140.