



HAL
open science

Etude du passage à l'échelle d'un système pair-à-pair de diffusion télévisuelle

Cyril Cassagnes, Damien Magoni, Chang Hyunseok, Sugih Jamin

► **To cite this version:**

Cyril Cassagnes, Damien Magoni, Chang Hyunseok, Sugih Jamin. Etude du passage à l'échelle d'un système pair-à-pair de diffusion télévisuelle. 11èmes Journées Doctorales en Informatique et Réseaux, Mar 2010, Sophia Antipolis, France. inria-00468050

HAL Id: inria-00468050

<https://hal.inria.fr/inria-00468050>

Submitted on 29 Mar 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Etude du passage à l'échelle d'un système pair-à-pair de diffusion télévisuelle

Cyril Cassagnes
University of Bordeaux
Bordeaux, France
cassagne@labri.fr

Damien Magoni
University of Bordeaux
Bordeaux, France
magoni@ieee.org

Hyunseok Chang
Bell Labs
Holmdel, NJ, USA
hyunseok@ieee.org

Sugih Jamin
University of Michigan
Ann Arbor, MI, USA
jamin@eecs.umich.edu

Résumé—La télévision transmise par le protocole Internet (IPTV) présente de nombreuses possibilités tant pour les utilisateurs que pour les fournisseurs de services. Elle a attiré de manière significative les entreprises et depuis quelques années, la communauté de la recherche. Parmi les architectures de diffusion de l'IPTV, le mécanisme de diffusion TV pair-à-pair est intéressant du fait de sa facilité à être déployé. Cependant, la question de comment une application P2PTV peut supporter un nombre croissant d'utilisateurs n'a pas été étudié en profondeur. Dans cet article, nous essayons d'aborder ces questions en étudiant la possibilité du passage à l'échelle et le facteur d'efficacité d'un réseau P2P diffusant de la vidéo temps réel. À travers l'utilisation de données fournies par un système de production P2PTV, nous avons fait des simulations dont les résultats montrent qu'il y a encore des difficultés à surmonter avant qu'un réseau P2PTV puisse devenir largement utilisé.

I. INTRODUCTION

Avec l'accroissement de la largeur de bande passante et les continuelles améliorations dans les technologies de compression vidéo, le service de télévision par internet (IPTV) connaît ces derniers temps, un maintien de sa croissance. Pour réaliser le service IPTV dans l'internet d'aujourd'hui, le mécanisme de diffusion basé sur le pair-à-pair (P2P) est considéré comme une option intéressante de part sa facilité à être déployé et pour le faible besoin de bande passante.

Dans un réseau P2P construit pour diffuser de la TV sur IP, les clients accèdent aux flux vidéos en se connectant au serveur de diffusion ou à un autre client présent sur le réseau. Le serveur de diffusion génère un flux vidéo par l'encodage du signal TV capturé depuis le satellite. Après s'être joints au réseau, les clients peuvent contribuer avec leur débit ascendant (*upload*) en transférant le flux vidéo entrant aux clients se joignant au réseau par la suite. Pour permettre une utilisation plus efficace de la capacité d'envoi des clients, les flux vidéos sont distribués via un réseau recouvrant (*overlay*) en morceaux (*chunks*), (e.g. [1]) ou sous-flux (e.g., [2], [3]). Les *chunks* sont des segments divisés en fonction du temps du flux vidéo, tandis que les sous-flux sont issus de la division spatiale du flux original (e.g., couches en H.264 SVC). Les *chunks* ou les sous-flux sont soit poussés par les clients amonts pour le transfert aux autres clients avals, soit tirés des clients amonts par les clients en aval. Cela dépend du protocole de partage utilisé dans le réseau P2P. Dans la distribution *pull-driven*, les clients cherchent et récupèrent un flux individuel, ils sont dans une ap-

proche opportuniste, tandis que dans l'approche *push-driven*, un client établit une connexion virtuelle de transmission vers un client, et continue à recevoir les données tirées depuis un transmetteur jusqu'à la terminaison de la connexion. Dans des travaux récents [4], la distribution avec la technique du *pull-push* est présentée comme plus efficace que son homologue *pull-based*.

Comparé au P2P traditionnel partageant des données ou à la diffusion (*streaming*) progressive pour la vidéo à la demande (VoD), l'expérience d'optimisation de l'utilisateur final dans un environnement P2PTV est une tâche non-triviale à cause des contraintes de délai et de mémoire tampon. De plus, les contraintes de capacité d'envoi et les comportements d'attrition inhérents aux clients participants peuvent s'ajouter à la difficulté de réaliser un système de diffusion extensible. Motivé par notre étude précédente sur l'adaptabilité opérationnelle du P2PTV [5], nous explorons l'impact de la distribution sur la bande passante d'envoi et les algorithmes de sélection d'un pair sur la performance du système. Dans notre étude, nous nous concentrons sur une architecture de *streaming* utilisant le *push-driven* et des sous-flux. Nous exécutons des simulations avec les données fournis par un système de production implémentant une telle architecture.

Le reste de ce papier est organisé comme suit. La Section II décrit le travail précédent fait sur un P2PTV. La Section III discute du passage à l'échelle de la redistribution dans des réseaux P2PTV. La Section IV propose des mécanismes pour l'efficacité croissante de l'algorithme de sélection de pair dans des réseaux P2PTV. La Section V présente des résultats d'évaluation obtenus en simulant l'architecture d'un des plus grand producteur fournissant le P2PTV en Europe. Nous concluons le papier en présentant des directions de recherche futures dans la dernière section.

II. TRAVAUX ANTÉRIEURS

Ces dernières années, beaucoup de système de diffusion multimédia ont été proposé et évalué dans la communauté de la recherche [6], [7], [8], [9], [10], [11]. En plus des prototypes de recherche, un certain nombre de systèmes P2PTV entièrement opérationnels sont apparus (e.g., PPLive, PPStream, SopCast, TVAnts, LiveStation, Joost et Zattoo). Avec la popularité grandissante du P2PTV, un grand nombre de papiers présentant des résultats de tests de systèmes P2PTV ont été publié,

abordant la résistance à la charge [12], les topologies [13] ou l'étude de systèmes spécifiques [14], [15], [16], [17], [18], [19]. De nombreuses propositions pour l'amélioration des systèmes P2PTV existant ont aussi été présentées, de la limite de faisabilité [20] et de la capacité d'envoi [21] à la conscience de la localité [22] et même à la théorie des fluides stochastiques pour des systèmes de *streaming* P2P [23]. Beaucoup de propositions récentes basées sur des motivations de robustesse [24], d'altruisme [25], de conscience de contribution [26] et de sous-flux négociant [27] visent à éviter les clients profiteurs dans des systèmes à grande échelle.

III. REDISTRIBUTION DANS UN RÉSEAU RECOUVRANT P2PTV

Dans cette section, nous étudions l'impact de la bande passante du client sur le passage à l'échelle d'un système P2P (i.e., le nombre maximal des utilisateurs qui peuvent être supportés par le système). Nous ne considérons pas de pair dynamique (e.g., joignons/enlevons des comportements). Nous ajouterons des pairs dynamiques dans des sections postérieures. Nous définissons le ratio du flux entrant qui peut être redistribué à d'autres pairs comme un *facteur de redistribution* et le nommons k partout dans ce papier. Le facteur de redistribution k peut varier de 0 à l'infini et peut être fractionner, selon la capacité de connexion du pair. Si $k = 1$, cela signifie que le pair peut redistribuer le flux complet. Si $k = 2$, cela signifie que le pair redistribue deux copies du flux. Si $k = 0,5$, cela signifie que le pair redistribue seulement la moitié du flux en raison des contraintes de la bande passante de sa connexion. Des valeurs fractionnaires sont possibles parce qu'un flux est divisé dans des sous-flux multiples, qui permettent à un pair de redistribuer seulement un sous-ensemble du flux. Techniquement, le facteur de redistribution peut être évalué comme le débit ascendant (*upload bitrate*) divisé par le débit descendant (*download bitrate*) à n'importe quel instant donné. Dans ce papier, nous supposons que le facteur de redistribution reste constant dans le temps pour n'importe quel pair.

A. Analyse théorique

D'abord, nous étudions d'un point de vue théorique l'impact du facteur de redistribution lors du passage à l'échelle. Comme le réseau recouvrant est un graphe acyclique orienté, nous pouvons définir la profondeur d'un pair dans le réseau recouvrant comme le nombre des liens du réseau recouvrant entre lui-même et les pairs connectés à la source. Si nous supposons que tous les pairs ont un même k , nous pouvons déterminer le nombre maximal de pairs seulement en utilisant le facteur de redistribution k , la profondeur du graph n et la capacité de la source C . Car le nombre total de pairs est égal à la somme d'une série géométrique. En effet, si U_n représente le nombre de pairs à la profondeur n et si $U_0 = C$, alors nous avons $U_n = k \times U_{n-1}$. Le nombre total de pairs est ainsi égal à :

$$S_n = \sum_{i=0}^n U_i = C \times \frac{1 - k^{n+1}}{1 - k} \quad (1)$$

Le nombre maximum de pair pouvant se connecter au système dépendront de la valeur de k :

- Si $k < 1$ alors le nombre de pairs convergera vers $\sum U_n \rightarrow \frac{C}{1-k}$ et donc le système ne passera pas à l'échelle en taille.
- Si $k = 1$ alors le nombre de pairs divergera de manière linéaire; $\sum U_n \rightarrow +\infty$, et donc le système passera à l'échelle en taille.
- Si $k > 1$ alors le nombre de pairs divergera de manière exponentielle; $\sum U_n \rightarrow +\infty$, et donc le système passera à l'échelle en taille.

Considérons un cas plus général où un niveau l est donné, chaque pair i a une capacité d'envoi (ul_i) et de téléchargement (dl_i) alors le facteur k pour ce niveau l contenant p pairs serait :

$$k_l = \frac{\sum_{i=1}^p ul_i}{\sum_{i=1}^p dl_i} \quad (2)$$

Nous définissons un pair inactif qui ne téléchargera pas la totalité du flux. Nous définissons un pair hyperactif comme ayant $k > 1$. Dans le cas le plus général, le facteur k de redistribution est exprimé comme ceci :

$$k = \frac{U_a \times (1 - R_i - R_h) + U_i \times R_i + U_h \times R_h}{D_a \times (1 - R_i - R_h) + D_i \times R_i + D_a \times R_h} \quad (3)$$

- D_a : nombre de téléchargement de sous-flux par pair actif
- U_a : nombre d'envoi de sous-flux par pair actif
- D_i : nombre de téléchargement de sous-flux par pair inactif
- U_i : nombre d'envoi de sous-flux par pair inactif
- U_h : nombre d'envoi de sous-flux par pair hyperactif
- R_i : ratio de pairs inactifs vs. pairs total
- R_h : ratio de pairs hyperactif vs. pairs total

Dans un monde idéal, nous aurions $k \geq 1$ pour tous les pairs et donc nous n'aurions pas de réseau recouvrant de taille limitée (bien que cela puisse créer un réseau recouvrant trop profond si k est à peine plus grand que 1). En réalité, nous avons un ensemble de facteur de redistributions assez variés.

- k peut être inférieur à 1
 - Profiteurs ($k = 0$)
 - Pairs bloqué par sécurité ($k = 0$)
 - Pairs avec une connexion peu performante ($k < 1$)
- k peut être supérieur à 1
 - Sources auxiliaire, i.e., super pairs ($k \gg 1$)
 - Pairs avec une connexion performante ($k > 1$)
 - Pair inactif, i.e., pair qui ne regarde pas le flux mais redistribue un sous ensemble du sous-flux du flux ($k > 1$ pour plusieurs sous-flux).

B. Les pairs inactifs

Pour rehausser un facteur k bas, une solution est de tirer parti des pairs inoccupés qui peuvent redistribuer le flux en arrière plan. Ne regardant pas le flux activement, les pairs inactifs pourraient recevoir seulement une fraction du flux complet et redistribuer les copies de cette fraction, agissant ainsi comme un multiplicateur. La figure 1 montre le réseau P2PTV en fonction du pourcentage de pairs inactifs. Le

nombre de sous-flux contenus dans le flux complet est mis à 16 comme dans le réseau recouvrant de Zattoo [5]. Il est assumé que des pairs actifs ont $k = 0.5$ et que des pairs inactifs redistribuent un sous-flux (1/16ème du flux) quatre fois (ainsi $k = 4$). Pour permettre une même disponibilité des différents sous-flux dans le réseau, chaque pair actif/inactif choisit aléatoirement le(s) sous-flux à redistribuer. Les courbes de la figure sont obtenues en utilisant la formule 3.

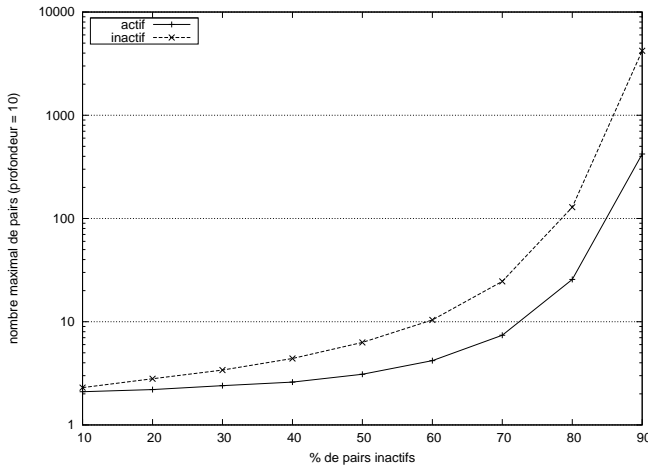


FIG. 1. Capacité du réseau vs. quantité de pairs inactifs.

La capacité du réseau est égale au nombre maximal de pairs du réseau recouvrant divisé par le nombre maximal des pairs qui peuvent se connecter à la source directement. Si la capacité du réseau est égale à 1, cela signifie qu'un pair peut seulement se connecter à la source, mais pas aux autres pairs. Si la capacité du réseau est égale à 10, cela signifie que le réseau recouvrant peut supporter dix fois plus de pairs que la capacité de la source. La figure montre à fois les capacités du réseau actif et du réseau inactif. La courbe étiquetée "actif" est le nombre de pairs dans le réseau avec $k = 0.5$. La courbe étiquetée "inactif" est la somme du nombre de pairs inactifs et de pairs actifs. Elle montre que des pairs inactifs aident très peu à améliorer la capacité du réseau quand des valeurs réalistes du nombre de pair inactif sont prises (<30%). Les pairs actifs d'un réseau recouvrant ont besoin d'un grand nombre de pairs inactifs pour s'étendre (c'est-à-dire avoir une capacité > 10). Ainsi l'utilisation de pairs inactifs n'aide pas le système à passer à l'échelle de manière significative.

C. Les pairs hyperactifs

Une autre solution pour rehausser un facteur k bas est d'utiliser des pairs hyperactifs pour redistribuer des flux multiples. Un pair hyperactif est un pair qui regarde un flux et peut redistribuer des copies multiples du flux, ou au moins un flux complet plus quelques sous-flux supplémentaires. La figure 2 montre la capacité du réseau de *streaming* en fonction du pourcentage de pairs hyperactifs. Semblable au cas précédent, un flux complet consiste en 16 sous-flux. Il est assumé que des pairs actifs régulièrement ont un $k = 0.5$ et que des pairs hyperactifs ont un $k = 1.5$. Les courbes de

la figure 2 sont aussi obtenues par utilisation des formules 3. La courbe étiquetée "hyperactif" est le nombre de pairs dans le réseau avec $k = 1.5$. La courbe étiquetée "actif" est la somme des nombres de pairs hyperactifs et de pairs actifs. Nous pouvons voir qu'il y a une divergence à 60% des pairs hyperactifs. Ainsi l'utilisation de pairs hyperactifs n'aide pas le système à passer à l'échelle de manière significative. Avec des pairs exclusivement hyperactifs la capacité du réseau croît exponentiellement alors que si il y a des pairs moins hyperactifs, ils doivent avoir des valeurs de k plus grande que 1.5 pour atteindre la divergence.

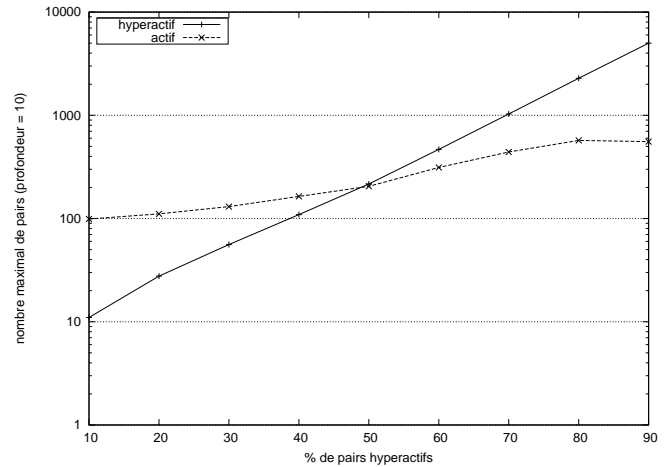


FIG. 2. Capacité du réseau vs. quantité de pairs hyperactifs.

IV. EFFICACITÉ DE L'ALGORITHME DE SÉLECTION D'UN PAIR DANS UN RÉSEAU P2PTV

Quand un nouveau pair se connecte au réseau recouvrant, il exécute son propre algorithme de sélection pour choisir les pairs auxquels se connecter. Si la sélection du pair est faite aléatoirement, le réseau résultant peut devenir inefficace pour deux raisons. D'abord, le réseau peut devenir trop profond et non assez large, engendrant ainsi de gros retard de lecture. Deuxièmement, le réseau peut subir beaucoup d'attrition, entraînant de nombreuses interruptions pour les utilisateurs.

A. Les effets sur le réseau recouvrant

Nous voudrions voir s'il est possible d'améliorer l'efficacité d'un réseau recouvrant P2PTV en utilisant des algorithmes de sélection de pairs qui fusionnent des paramètres dynamiques tels que la capacité d'envoi, des temps de sessions, des distances entre des pairs et des positions de profondeur dans le réseau recouvrant. L'idée derrière cette proposition est qu'en plaçant des pairs plus stable et avec une bande passante à forte capacité tout près de la source, on puisse rendre le réseau recouvrant plus efficace. L'algorithme de sélection de pair est idéal pour influencer l'évolution du réseau car nous pouvons contrôler où les pairs se connecteront dans le réseau. Si l'algorithme de sélection de pair réussit à placer des pairs stables près de la source, cela doit réduire les réorganisations complètes du réseau recouvrant. Aussi, si cet

algorithme réussit à placer des pairs avec une bande passante à forte capacité près de la source, cela devrait augmenter la capacité du réseau recouvrant en gardant une profondeur raisonnable. Nous étudierons cette approche par la simulation dans la Section V.

B. Les effets sur un pair

En réalité, les pairs changent de canaux et redistribuent seulement un minimum de flux dans le but d'économiser leur bande passante d'envoi limitée. De tels comportements sont nuisibles à la croissance d'un réseau recouvrant et donc au passage à l'échelle. Ainsi quelle peut être la motivation pour le pair d'un P2PTV de se comporter autrement? Une solution est d'employer le mécanisme "d'un prêté pour un rendu" avec des vidéos de qualités différentes comme proposé dans [28]. Si différentes qualités de vidéos ne sont pas disponibles, une autre solution que nous proposons permet de négocier la position de profondeur dans le réseau recouvrant. Si le pair a une bonne connexion et reste dans un même canal, il deviendra progressivement plus proche de la source (mieux servi). À la fin, les pairs avec une connexion stable et performante seraient exposés à moins de variations de leur connectivité et jouiraient de la TV en temps réel. Ils seraient donc naturellement incité à rester dans un même canal. On a montré dans [16] que le retard de lecture dans un système P2PTV à grande échelle varie entre 30 et 90 secondes. Ainsi, la position d'un pair pourrait être une bonne motivation pour inciter un pair stable et performant à rester sur un même canal. Dans les résultats de simulation montrés dans la Section V, nous utilisons ces critères (e.g., le temps de session et la capacité d'envoi) pour voir si le réseau fonctionne mieux quand les pairs sont choisis en fonction de tels critères.

V. SIMULATION D'UN RÉSEAU RECOUVRANT P2PTV

Après avoir étudié quelques aspects théoriques d'un réseau recouvrant P2PTV et proposé des mécanismes d'amélioration, nous présentons maintenant des résultats obtenus par simulation et mettons en évidence l'impact des algorithmes de sélection de pair sur la performance d'un système de P2PTV.

A. Paramètres et métriques de la simulation

Notre code de simulation met en oeuvre la technique des sous-flux *push-driven* de Zattoo basé sur une architecture de P2PTV [3] et a été exécuté sur le logiciel de simulation réseau *nem*¹. Une carte Internet de 4.2k-noeuds a été utilisée comme topologie sous-jacente. Nous supposons qu'un réseau donné distribue une chaîne de télévision, et nous ne prenons pas en considérations les variations de taille de canal quotidienne. Chaque test dure 12 heures et seulement les 6 dernières heures sont analysées. Après la 6ème heures, nous sommes dans un régime stable d'état. Chaque valeur de résultat est la moyenne de 30 tests et les valeurs d'écart-type sont fournies.

Pour toutes nos simulations, nous utilisons les mêmes paramètres que dans notre travail précédent [5], qui était basé sur l'analyse de 9.6M de sessions enregistrées par le P2PTV

Zattoo pendant une période de 2 semaines. Selon les données de session, la fonction de distribution cumulative (CDF) du facteur de redistribution k suit une distribution exponentielle. 50% des pairs peuvent redistribuer moins de 50% du flux complet (i.e., $k < 0.5$). 82% des pairs peuvent redistribuer moins que le flux complet (i.e., $k < 1$). La capacité de la connexion des pairs individuels est assignée pour que la distribution résultante devienne la même que la distribution empirique. Le type de NAT d'un pair, qui détermine son accessibilité au réseau, est aussi considéré dans la distribution empirique des types de NAT énoncés dans [5]. Finalement, le temps d'inter-arrivée dans la session et le temps de session sont tout deux des distributions exponentielles équivalentes énoncées dans [5]. Nous faisons varier la taille du canal du mode famine (proche de la capacité source) au mode profusion (plusieurs fois supérieur à la capacité source). La table I montre les paramètres de maintien saisie.

TABLE I
PARAMETRES DE SIMULATION

Paramètres	Valeurs
Nombre de sous-flux par flux)	16
Capacité de la source	50 clients
Période de recherche	2 sec
Maximum de tentatives de recherche	2
Taille maximale de la liste de pairs candidats	40 pairs
Nombre de simulations par scénario	30

Pour évaluer le fonctionnement d'un système de P2PTV, nous étudions les métriques suivantes :

- 1) Ratio du temps de visualisation (en % = $100 \times$ temps de visualisation du pair / durée de vie du pair) : le temps de visualisation moyen des pairs qui ont terminé pendant cette période.
- 2) Pourcentage de pairs rejetés : nombre total des pairs qui ne pouvaient pas se connecter au réseau recouvrant P2PTV pendant la période donnée divisé par le nombre de nouveaux pairs par période.
- 3) Nombre moyen d'interruptions par pair : nombre total d'interruption vidéo vu pour tous les pairs dans la période donnée divisé par le nombre de nouveaux pairs par période.

Nous analysons ces métriques de production en variant le nombre de nouveaux pairs arrivant par heure, qui définit la charge mis sur la source et le réseau P2P.

B. Résultats de simulation

Cette section présente nos résultats de simulation. Un pair essayant de se connecter à d'autres pairs pour obtenir tous les sous-flux nécessaires est appelé un pair orphelin. Il envoie des messages de recherche pour découvrir d'autres pairs, envoie des messages d'appairage pour se connecter aux pairs disponibles et obtient finalement le flux complet de leur part. Un pair qui est capable, et disposé à offrir une partie de tous

¹<http://www.labri.fr/perso/magoni/nem/>

les sous-flux demandés par un pair orphelin est appelé pair adoptif. Un pair adoptif est un pair qui a positivement répondu au message de recherche d'un orphelin. Après avoir eu de multiples réponses positives des pairs adoptifs candidats, Les orphelins doivent choisir le pair auquel il devrait envoyer un message pour se connecter. Nous évaluons les algorithmes de sélection des pairs suivants, décrit dans la Section IV :

- **Aléatoire** : Un pair orphelin essaie de se connecter à un pair adoptif sélectionné aléatoirement
- **Local** : Un pair orphelin essaie de se connecter à un pair adoptif proche (la distance étant mesuré en sauts)
- **Débit ascendant (upload)** : Un pair orphelin essaie de se connecter à un pair adoptif proposant le meilleur envoi en amont (mesuré dans sur le nombre de sous-flux)
- **Durée d'activité** : Un pair orphelin essaie de se connecter à un pair adoptif ayant la plus longue durée d'activité.

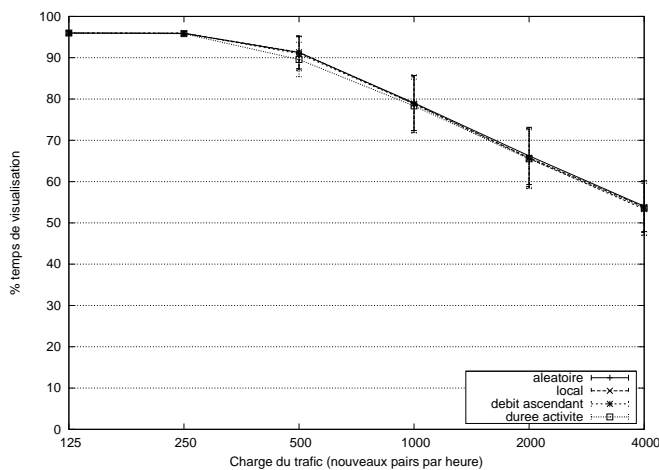


FIG. 3. Temps de visualisation moyen vs. charge de trafic.

La figure 3 montre le temps moyen de visualisation d'un pair en fonction de la charge du trafic. Nous pouvons voir que les effets des divers algorithmes sur le temps d'observation ne font pas beaucoup de différence comparé à une sélection aléatoire. Bien qu'il y ait 7% de différence entre le plus mauvais et les meilleurs algorithmes pour une charge de trafic de 2000 nouveaux pairs par heure et 5% de différence pour une charge de 4000, ces valeurs ne sont pas énormes. Ce résultat quelque peu inattendu induit la relative importance des paramètres de niveau utilisateur telle que la capacité d'envoi et le temps de session sur les paramètres systèmes tels que les algorithmes de sélection de pair.

La figure 4 montre que le pourcentage moyen de pairs rejeté est fonction de la charge de trafic. En comparant avec les résultats précédents, la diminution en temps de visualisation est principalement dûe aux pairs étant rejeté du réseau. Seulement un petit pourcentage est causé par les interruptions dûes aux déconnexions et reconnexions de pairs.

La figure 5 montre le nombre moyen d'interruptions par pair en fonction de la charge de trafic. Nous observons que le nombre moyen d'interruptions par pair augmente progressivement quand le nombre de pairs augmente. Quand le nombre

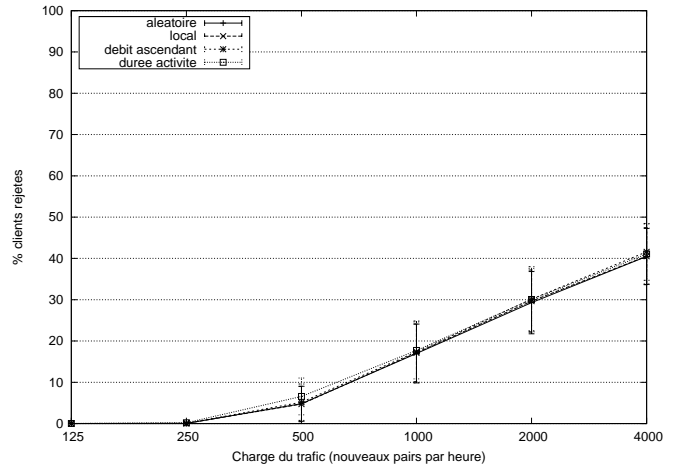


FIG. 4. Pourcentage moyen de pairs rejetés vs. charge de trafic.

de nouveau pairs augmente, le réseau recouvrant grandit et le taux d'attrition moyen devient plus grand et ainsi mène à plus de connexions et reconnexions. Cependant, quand le réseau recouvrant devient saturé par les nouveaux pairs, ces nouveaux pairs ne peuvent pas réussir à se joindre au réseau et sont rejeté. Ainsi, le nombre de connexions et de reconnexions ne grandit pas tant, car les pairs rejetés ne contribuent pas significativement à ce nombre. Cependant, le nombre total de pairs augmente toujours, et donc la proportion n'augmente plus désormais.

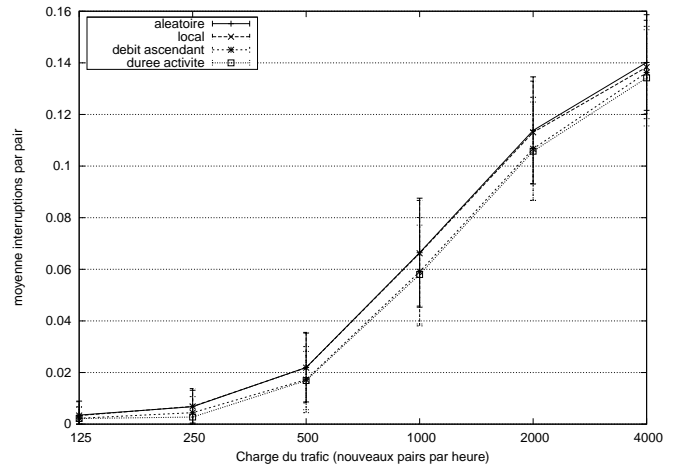


FIG. 5. Nombre moyen d'interruptions par pair vs. charge de trafic.

Deux remarques peuvent expliquer les résultats des algorithmes alternatifs par rapport à l'algorithme de sélection aléatoire. D'abord, les sessions sont de durée limitée. Environ 50% des sessions sont plus brèves que 1,5 minutes. Cela crée beaucoup d'attrition qui rendent l'évolution du réseau recouvrant difficile à contrôler au cours du temps. Deuxièmement, le facteur de redistribution tend fortement vers de petites valeurs. 50% des pairs ont une capacité d'envoi inférieure à 50%. Ainsi, les pairs avec de long temps session

peuvent avoir une faible capacité d'envoi et ne pas être trop utilisés. En somme, ces deux facteurs pèsent beaucoup plus lourdement sur le temps de visualisation que les algorithmes de sélection divers. Quand la charge de trafic est grande, l'algorithme qui fonctionne mieux que la sélection aléatoire est l'algorithme "local". Le résultat des simulations avec chacune des métriques définies prouve que l'algorithme de sélection n'affecte pas beaucoup le système.

VI. CONCLUSION

Dans ce papier nous avons étudié le comportement d'un réseau P2PTV *push-driven* par sous-flux. Nous avons conduit une étude théorique concernant l'impact du facteur de redistribution et avons montré que le passage à l'échelle du système dépend fortement de la capacité d'envoi disponible dans les pairs. Nous avons aussi montré que l'utilisation de pairs inactifs aussi bien que de pairs hyperactifs n'est pas une solution fondamentale pour que le système passe à l'échelle. Nous avons proposé l'amélioration possible du processus de sélection de pair étant donné que le processus de sélection a un impact direct sur la structure et l'efficacité du réseau recouvrant. Pour évaluer son impact potentiel, nous avons effectué des expériences de simulation pour mesurer l'efficacité des divers algorithmes de sélection des pairs dans un système P2PTV avec une forte charge. Nous avons effectué notre simulation avec des paramètres réalistes tirés des données de 9.8M de sessions rassemblées par le système P2PTV de niveau professionnel Zattoo. Nos résultats montrent que le facteur de redistribution et le temps de session ont un effet profond sur la capacité maximale du réseau recouvrant P2PTV et que les divers algorithmes de sélection jouent un rôle relativement marginal pour le passage à l'échelle du système. Nos travaux futurs seront destinés à l'étude d'autres algorithmes de sélection de pair ainsi qu'à l'étude de l'impact de la taille des mémoires tampons et des paramètres de recherche de pairs sur le rendement global d'un système P2PTV.

RÉFÉRENCES

- [1] L. Bracciale, F. L. Piccolo, D. Luzzi, S. Salsano, G. Bianchi, and N. Blefari-Melazzi, "A Push-based Scheduling Algorithm for Large Scale P2P Live Streaming," in *Proceedings of the 4th International Telecommunication Networking Workshop on QoS in Multiservice IP Networks*, 2008, pp. 1–7.
- [2] S. Xie, B. Li, G. Y. Keung, and X. Zhang, "CoolStreaming : Design, Theory, and Practice," *IEEE Trans. on Multimedia*, vol. 9, no. 8, December 2007.
- [3] H. Chang, S. Jamin, and W. Wang, "Live Streaming Performance of the Zattoo Network," in *Proceedings of Internet Measurement Conference*, November 2009.
- [4] M. Zhang, Q. Zhang, and S.-Q. Yang, "Understanding the Power of Pull-based Streaming Protocol : Can We Do Better ?" *IEEE JSAC*, vol. 25, no. 9, pp. 1678–1694, 2007.
- [5] K. Shami, D. Magoni, H. Chang, W. Wang, and S. Jamin, "Impacts of Peer Characteristics on P2PTV Networks Scalability," in *Proceedings of the 28th IEEE Conference on Computer Communications – Mini-Conference*, April 2009.
- [6] D. Tran, K. Hua, and T. Do, "ZIGZAG : An Efficient Peer-to-Peer Scheme for Media Streaming," in *Proceedings of the IEEE Infocom*, 2003.
- [7] R. Rejaie and S. Stafford, "A Framework for Architecting Peer-to-Peer Receiver-Driven Overlays," in *Proceedings of the ACM NOSSDAV*, 2004, pp. 42–47.
- [8] X. Liao, H. Jin, Y. Liu, L. Ni, and D. Deng, "Anysee : Peer-to-Peer Live Streaming," in *Proceedings of the IEEE Infocom*, 2006.
- [9] X. Zhang, J. Liu, B. Li, and T.-S. Yum, "CoolStreaming/DONet : A Data-driven Overlay Network for Live Media Streaming," in *Proceedings of the 24th IEEE Infocom*, 2005, pp. 2102–2111.
- [10] F. Pianese, "PULSE : A Flexible P2P Live Streaming System," in *Proceedings of the 9th IEEE Global Internet Symposium*, 2006.
- [11] F. Pianese, D. Perino, J. Keller, and E. Biersack, "PULSE : An Adaptive, Incentive-Based, Unstructured P2P Live Streaming System," *IEEE Transactions on Multimedia*, vol. 9, no. 6, 2007.
- [12] K. Sripanidkulchai, B. Maggs, and H. Zhang, "An Analysis of Live Streaming Workloads on the Internet," in *Proceedings of the ACM IMC*, 2004, pp. 41–54.
- [13] C. Wu, B. Li, and S. Zhao, "Magellan : Charting Large-Scale Peer-to-Peer Live Streaming Topologies," in *Proceedings of the 27th International Conference on Distributed Computing Systems (ICDCS'07)*, 2007, p. 62.
- [14] M. Cha, P. Rodriguez, S. Moon, and J. Crowcroft, "On Next-Generation Telco-Managed P2P TV Architectures," in *Proceedings of the 7th International Workshop on Peer-to-Peer Systems*, 2008.
- [15] S. Xie, G. Keung, and B. Li, "A Measurement of a Large-Scale Peer-to-Peer Live Video Streaming System," in *Proceedings of the International Conference on Parallel Processing Workshops (ICPPW'07)*, 2007, p. 57.
- [16] X. Hei, C. Liang, J. Liang, Y. Liu, and K. Ross, "A measurement study of a large-scale p2p iptv system," *IEEE Transactions on Multimedia*, vol. 9, no. 8, 2007.
- [17] L. Vu, I. Gupta, J. Liang, and K. Nahrstedt, "Measurement and Modeling of a Large-scale Overlay for Multimedia Streaming," in *The Fourth International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness*, 2007.
- [18] S. Ali, A. Mathur, and H. Zhang, "Measurement of Commercial Peer-to-Peer Live Video Streaming," in *Proceedings of the ICST Workshop on Recent Advances in Peer-to-Peer Streaming*, 2006.
- [19] G. Marfia, G. Pau, P. D. Ricco, and M. Gerla, "P2P Streaming Systems : A Survey and Experiments," in *Proceedings of the STreaming Day*, 2007.
- [20] K. Sripanidkulchai, A. Ganjam, B. Maggs, and H. Zhang, "The Feasibility of Supporting Large-Scale Live Streaming Applications with Dynamic Application End-points," in *Proceedings of the ACM SIGCOMM*, 2004.
- [21] J. Douceur, J. Lorch, and T. Moscibroda, "Maximizing Total Upload in Latency-Sensitive P2P Applications," in *Proceedings of the 19th ACM SPAA*, 2007, pp. 270–279.
- [22] F. Pianese and D. Perino, "Resource and Locality Awareness in an Incentive-based P2P Live Streaming System," in *Proceedings of the ACM P2P-TV*, 2007.
- [23] R. Kumar, Y. Liu, and K. Ross, "Stochastic Fluid Theory for P2P Streaming Systems," in *Proceedings of the IEEE Infocom*, 2007.
- [24] M. Feldman, K. Lai, I. Stoica, and J. Chuang, "Robust Incentive Techniques for Peer-to-Peer Networks," in *Proceedings of the 5th ACM Conference on E-Commerce*, 2004, pp. 102–111.
- [25] Y. Chu and H. Zhang, "Considering Altruism in Peer-to-Peer Internet Streaming Broadcast," in *Proceedings of the ACM NOSSDAV*, 2004.
- [26] Y.-W. Sung, M. Bishop, and S. Rao, "Enabling Contribution Awareness in an Overlay Broadcasting System," in *Proceedings of the ACM SIGCOMM*, 2006.
- [27] Z. Liu, Y. Shen, K. Ross, S. Panwar, and Y. Wang, "Substream Trading : Towards an Open P2P Live Streaming System," in *Proceedings of the IEEE International Conference on Network Protocols*, 2008, pp. 94–103.
- [28] Z. Liu, Y. Shen, S. S. Panwar, K. W. Ross, and Y. Wang, "Using Layered Video to Provide Incentives in P2P Live Streaming," in *Proceedings of the ACM Workshop on Peer-to-peer streaming and IP-TV*, 2007, pp. 311–316.