Un mécanisme distribué de synchronisation appliqué à un cas de téléconsultation

Raffaele Noro
Ecole Polytechnique Fédérale de Lausanne, Suisse
Groupe Services de Télécommunications, Laboratoire TCOM
E-Mail : noro@tcom.epfl.ch

Résumé

Cet article présente la conception et l'implantation d'un mécanisme de synchronisation dans le cadre de la manipulation à distance d'images radiologiques et pathologiques.

1 Introduction

Dans le domaine des applications multimédia, on se trouve face à une demande de plus en plus importante de *Qualités de Service* (QoS) au niveau de l'utilisateur. La disponibilité de nouvelles technologies dans le cadre des réseaux à haut-debit, telles que ATM [1], avec la possibilité de spécification d'une certaine gamme de QoS, au niveau de la couche ATM, nous aide à répondre (partiellement) aux besoins engendrés par de telles applications.

Puisque, malheuresement, il n'y a pas des liaisons immé-diates entre cette QoS (bande passante, taux de pertes, temps de transfert) et la qualité que l'application doit offrir á l'utilisateur (taille des images, influence de pertes, degrée d'interactivite, ...), nous devons trouver les moyens pour fournir un service fiable, sur la base des QoS offertes par le réseau. Nous nous focalisons sur un aspect fondamental de la qualité: la *synchronisation* . L'approche habituelle de la synchronisation [2] est de contrôler et de gérer la présentation des deux (ou plusieurs) flux concurrents assez proche de l'interface avec l'utilisateur (Figure 1). Dans cet article nous proposons et discutons un mécanisme distribué qui agit á plusieurs niveaux de l'infrastructure de transfert, de maniére á pouvoir manipuler deux flux en direction opposés et resynchroniser le système au point le plus approprié.

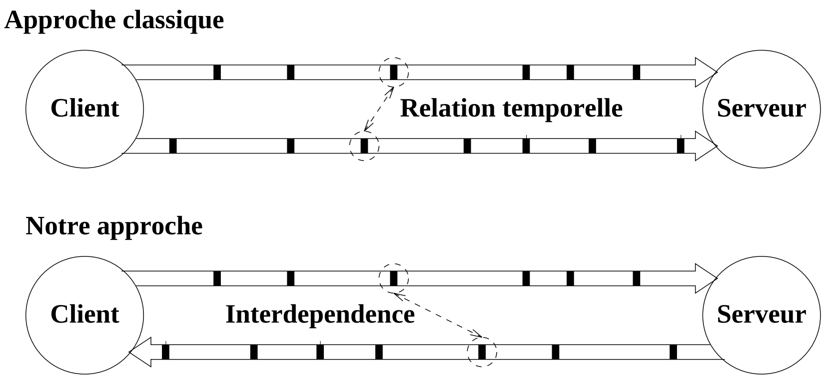

Figure 1: *Differentes approches à la synchronisation.*

2 Identification du problème et solution

Du point de vue de l'utilisateur final, la contrainte suivante doit être respectée:

> "A chaque action, le système doit réagir à une demande d'opération dans un intervalle de temps donné (toléré)".

Quantitativement [3], cela signifie que chaque réponse arrive dans un intervalle établi (par exemple 500 ms). Pour résoudre le problème, les étapes ci-dessous ont étés suivies:

- definition du concept de *stream*: un *stream* est un train d'*unités* avec des relations temporelles bien défi-nies. Notre système est basé sur le concept de *stream* (*stream* de control, *stream* MPEG, etc.).
- une *unité* représente une quantité de données qui appartiennent à un instant unique de temps: cet instant est defini à travers une indication de temps (*time stamp*) inserée dans le flux. Par exemple, une *unité de présentation* correspond à une image vidéo (*frame*), une *unité d'accès* , correspond à une image codée MPEG.
- definiton d'une hiérarchie entre *streams*: un *stream maître* et un *stream esclave* sont présent.
- La vérification est faite par un mécanisme de synchronisation adéquat au niveau du dispositif de présenta-tion (Figure 3, Point de vérification principal).

Certaines resources sont allouées sur le réseau (bande passante, ...) ainsi que dans les stations (buffers, codec, CPU, ...) pour obtenir (potentiellement) la synchronisation: cependant, si on laisse fonctionner librement le système, aucune garantie n'est donnée sur le maintien de l'état de synchronisation, ni sur la récupération de cet état dans le cas de pertes occasionelles de synchronisation. Par contre, si on contrôle en permanence le passage des données, on peut régler convenablement le flux, dans le but d'assurer une présentation synchrone.

Le mécanisme de synchronisation en question génére une série adaptée de paramètres pour l'algorithme sous-jacent (Figure 2).

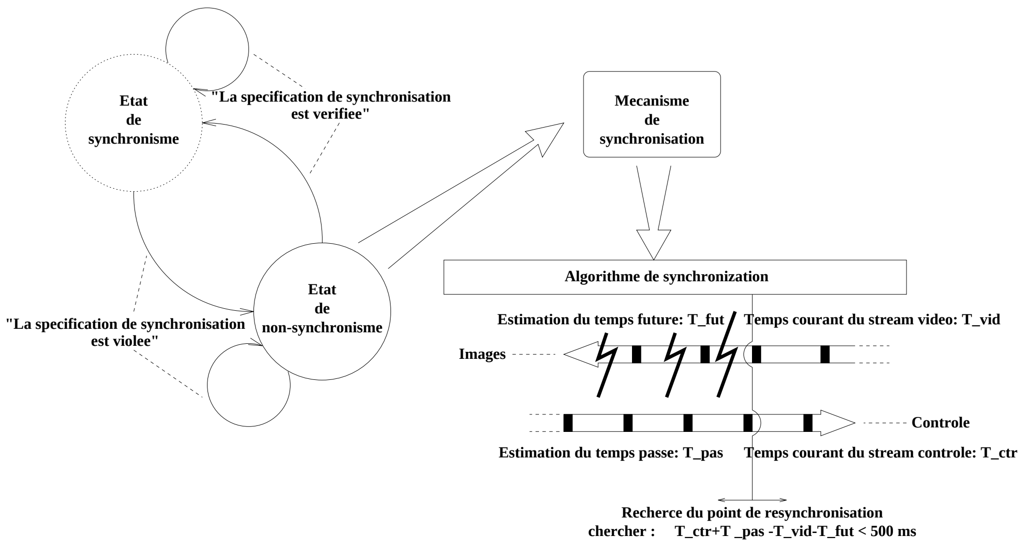

Figure 2: *Le mécanisme de synchronisation.*

L'algorithme est activé si et seulement si la spécification est violée au point de vérification principal: il examine, alors, les unités dans les *streams* (Figure 2). Connaissant le temps associé a chaque unité (*time stamp*), l'algorithme cherche, dans les deux flux qui se croisent, le premier point utile correspondant à la spécification de synchronisation. A ce stade, il est capable de re-synchroniser la présentation, en accélérant graduellement le flux de sortie. Pour faire fonctionner correctement l'algorithme [4], il est nécessaire d'estimer le temps que met une unité pour passer dans les blocs fonctionels de l'infrastructure (couche d'adaptation, files d'attente, decodage, ...). Ceci nous permet, ainsi, d'estimer le temps de présentation d'une unité avant qu'elle ne soit traitée (Figure 2).

3 Scenario

Un service de consultation à distance à été utilisé comme environement de test [5]. Il est ainsi réalisé: un *serveur* centralise les dossiers complets des patients (avec histoire clinique, données chirurgicales, images médicales, etc.). Le spécialiste accède à une partie de ce dossier, notamment aux images radiologiques ou pathologiques, à partir d'un poste de travail distant (station *client*), relié au serveur à travers une connexion ATM: cette connexion est, donc, destinée à transporter des données, dans un sens, et des fonctions de contrôle, dans l'autre.

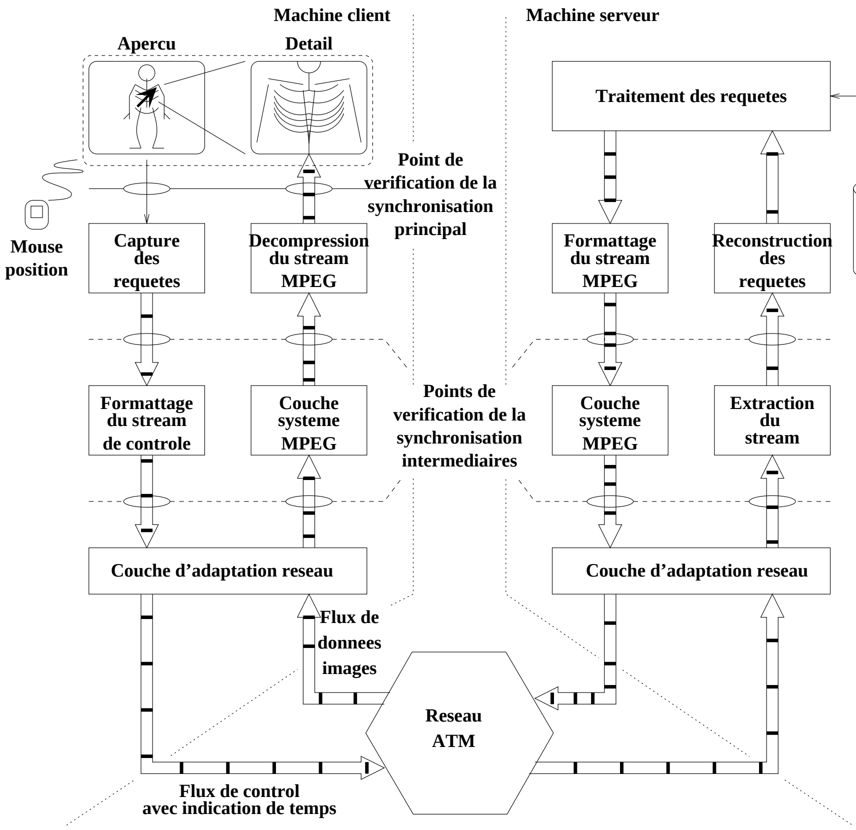

Figure 3: *Scénario d'une application de télémédecine.*

Le serveur est aussi conçu pour effectuer des traitements sur les images. L'utilisateur contrôle ces traitements à travers une interface graphique (menu-souris) qui transmet des requêtes au serveur (zoom, mesures de distance, mesures de niveau de gris, etc.). Comme il est probable qu'une séquence de requêtes soit générée par l'utilisateur, une sé-quence d'images va être renvoyée, en réponse, au client: la façon la plus efficace de transmettre de telles données est de les encapsuler en format MPEG [6] pour le transfert sur réseau ATM. Pour cette raison, la base de données contient les images à haute résolution, en format MPEG pré-compressé.

4 Resultats et conclusions

Notre mécanisme de synchronisation est capable d'assurer un service de synchronisation fiable au niveau de la présentation. Lorsque l'on arrive dans une situation "hors-synchronisation", on peut revenir à l'état normal dans un intervalle de 2-4 unités de présentation equivalent a 100-200 ms de vidéo (en fonction de la complexité de la scène). Contrairement à d'autres algorithmes de synchronization, on peut re-synchronizer sans devoir forcément renoncer à d'autres paramètres de qualité (intégrité et qualité de la séquence d'images, notamment); en effet, on est en mesure de recupérer le point de synchronisation au point optimal du flux.

Le service de téléconsultation offert est fiable: les premiers expériments ont été jugés très satisfaisants par les utilisateurs.

De plus, plusieurs applications distribuées peuvent tirer avantage d'un tel mécanisme de synchronisation: en effet, on est en train d'adapter le mécanisme à différent cas d'applications multimédia, où plusieurs machines clients interviennent et où des besoins plus importants en termes de synchronisation sont présents.

Références

[1] M. De Prycker, *Asynchronous transfer mode: Solution for broadband ISDN*, Ellis Horwood, Bodmin, Cornwall, GB, 1991.

[2] IEEE JSAC, *Synchronization issues in multimedia communications*, IEEE Communications society, Piscataway, NJ, USA, Jan. 1996.

[3] R. Steinmetz and K. Nahrstedt, *Fundamentals in Multimedia Computing and Communications*, Prentice - Hall, Englewood Cliffs, NJ, USA, 1995.

[4] L. Li, A. Karmouch, and N. D. Georganas, "Multimedia teleorchestra with independent sources: Part 2 - synchronization algorithms", *in ACM/Springer Multimedia Systems, vol. 1*, Feb. 1994.

[5] A. Basso, O. Verscheure, R. Noro, J.P. Hubaux, R. Meuli, R. Laurini, and P. Patthéy, "A multimedia architecture for medical tele-imaging over ATM", *in CAR '96*, Paris, France, June 1996.

[6] MPEG-2, *Information technology - Generic coding of moving pictures and associated audio information - Parts 1, 2*, ISO/IEC JTC 1, 1994.