



**HAL**  
open science

## Mise en oeuvre d'un modèle de corrélation QoS-QoE

Laouhari Benyettou

► **To cite this version:**

Laouhari Benyettou. Mise en oeuvre d'un modèle de corrélation QoS-QoE. Réseaux et télécommunications [cs.NI]. 2012. dumas-00725182

**HAL Id: dumas-00725182**

**<https://dumas.ccsd.cnrs.fr/dumas-00725182v1>**

Submitted on 24 Aug 2012

**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.

# RAPPORT DE STAGE

## Mise en œuvre d'un modèle de corrélation QoS-QoE

Étudiant : **BENYETTOU Lahouari**

Encadrant : **SOUIHI Sami**

Directeur de stage : **MELLOUK Abdelhamid**

**LiSSi : Laboratoire Images Signaux & Systèmes Intelligents**

Master 2 Recherche en Informatique  
Université de Bretagne Occidentale  
Parcours : Systèmes et Objets Communicants Mobiles



18 Juin 2012

# Résumé

Ces dernières décennies ont connu une énorme avancée technologique dans le domaine des réseaux de paquet. L'internet faisant partie de cette catégorie de réseaux, a vu naître de nombreuses applications multimédias qui l'utilisent comme moyen de diffusion. Divers services, tels que la télévision par internet, la vidéo à la demande, radio sur l'internet, la lecture de données multimédia, la téléphonie IP ou la téléconférence, sont devenues notre attraction quotidienne et représentent un secteur fleurissant du marché des télécommunications et un domaine de recherche actif ; particulièrement sur l'intégration efficace des services multimédias et l'exploitation intelligente des réseaux de paquet. En effet, ces services doivent offrir des qualités similaires à ceux des services de TV et de la téléphonie conventionnelle. Cependant, l'Internet n'a pas été conçue pour supporter le temps réel et les contraintes temporelles imposées par ce genre d'application d'une part, et d'autre part, il est peu commun que les utilisateurs soient satisfaits par des services de qualité inférieure à ceux dont ils ont pris l'habitude d'utiliser. Ainsi, la qualité de ce type de services imposant de telles contraintes, est fortement dépendante du contenu du service lui-même, de la capacité du réseau et des applications offrant et consommant le service. Plusieurs travaux [1,2, 3] montrent que certains paramètres vidéo, audio et réseau comme le débit, le taux de pertes, le délai et la gigue ont une influence sur la qualité perçue par les utilisateurs. Par ailleurs, il existe aussi d'autres facteurs liés à l'environnement externe qui affectent la qualité d'expérience utilisateur (QoE).

Dans le cadre du projet IPNQSIS (IP Network Monitoring for Quality of Service Intelligent Support) dont l'objectif est de développer des systèmes de surveillance continue et d'étudier le comportement de la qualité d'expérience (QoE) par l'analyse des performances du réseau et des services et leur impact sur les utilisateurs finaux. Notre tâche était d'automatiser le processus d'évaluation de la qualité perçue par les utilisateurs en temps réel, afin de pouvoir réagir aux changements spontanés des conditions du réseau. À cet effet, nous avons réalisé des expériences pour construire une base de données contenant les facteurs affectent la qualité vidéo et l'évaluation de la qualité par des sujets humains. Puis nous avons mis en corrélation les facteurs et la QoE en employant un réseau de neurones aléatoires (RNN : Random Neural Network) qui a déjà donné ses preuves dans l'estimation de la qualité audio dans [3]. Enfin, nous avons étudié l'impact des paramètres réseau sur la QoE.

# Table des matières

Chapitre 1 .....	- 1 -
1.1 Laboratoire de recherche: LISSI .....	- 1 -
1.2 Contextes .....	- 1 -
Chapitre 2 .....	- 2 -
Introduction .....	- 2 -
2.1 La Qualité d'Expériences Utilisateur .....	- 2 -
2.2 Évaluations de la QoE .....	- 3 -
2.3 Paramètres affectant la QoE .....	- 3 -
2.3.1. Paramètres réseau.....	- 3 -
2.3.2 Caractéristiques du contenu .....	- 4 -
2.3.3 Type du terminal .....	- 6 -
2.3.4 Facteurs psychologiques .....	- 6 -
Chapitre 3 .....	- 7 -
3.1. Objectifs .....	- 7 -
3.2 .Méthodologie .....	- 7 -
3.2.1 Mesure subjective de la qualité vidéo.....	- 8 -
3.2.1.1 Échelle de dépréciation a double stimulus (DSIS) .....	- 8 -
3.2.1.2 Méthode de stimulus unique .....	- 8 -
3.2.1.3 Méthode de comparaison de stimulus.....	- 8 -
3.2.1.5 Échelle continue de la qualité en double stimulus (DSCQS).....	- 9 -
3.2.1.6 Note absolue catégorie (ACR) .....	- 9 -
3.2.1.7 Note par catégories de dégradation (DCR).....	- 10 -
3.2.1.8 Méthode de comparaison par paires (PC).....	- 11 -
3.2.2 Analyse statistique de la MOS.....	- 11 -
3.2.3 Les réseaux de neurones aléatoires.....	- 12 -
3.2.3.1 Le neurone élémentaire dans RNN .....	- 12 -
3.2.3.2 Fonctionnement en réseau.....	- 13 -
3.2.3.3 Architecture des RNN.....	- 14 -
3.2.3.4 L'apprentissage des RNN.....	- 14 -
3.3. Expérience.....	- 15 -
3.3.1. Configuration de l'expérience .....	- 15 -
3.3.2 Choix des séquences vidéo .....	- 16 -
3.3.3 Choix des paramètres.....	- 16 -
3.3.4 Génération d'ensemble de données.....	- 17 -
3.3.5 Application .....	- 18 -
3.3.6 Le scénario expérimental du test subjectif .....	- 19 -
3.4. Résultats .....	- 21 -
3.4.1 Architecture du réseau de neurone .....	- 21 -
3.4.2 Comportement de réseaux de neurones .....	- 22 -
3.4.3 Impact des paramètres sur la qualité vidéo.....	- 23 -
3.4.3.1 Impact de LBW et de PER sur la qualité .....	- 23 -
3.4.3.2 Impact de SBR et LBW sur la qualité.....	- 24 -
3.4.3.3 Impact de la gigue et LBW sur la qualité.....	- 24 -
3.4.3.3 Impact de la gigue et PER sur la qualité .....	- 25 -
3.5 Conclusions .....	- 26 -

3.6 Perspectives .....	- 26 -
Chapitre 4 .....	- 28 -
4.1 Bilan technique.....	- 28 -
4.2 Bilan humain .....	- 28 -
4.3 Bilan du projet.....	- 28 -
Bibliographies .....	- 29 -
Annexe .....	- 31 -

# Table des figures

<i>Figure-1: QoE de bout-en-bout.....</i>	<i>- 3 -</i>
<i>Figure-2 : Variabilité du jugement qualitatif, en fonction de l'intérêt pour le contenu . ....</i>	<i>- 5 -</i>
<i>Figure-3 : Présentation du stimulus dans la méthode ACR.....</i>	<i>- 10 -</i>
<i>Figure-4 : Présentation du stimulus dans la méthode DCR .....</i>	<i>- 10 -</i>
<i>Figure-5 : Présentation du stimulus dans la méthode PC .....</i>	<i>- 11 -</i>
<i>Figure-6: Représentation du neurone de RNN .....</i>	<i>- 12 -</i>
<i>Figure-7 : Architecture des RNN .....</i>	<i>- 13 -</i>
<i>Figure-8: Configuration de l'expérience .....</i>	<i>- 16 -</i>
<i>Figure-9: Séquences vidéo choisies .....</i>	<i>- 16 -</i>
<i>Figure-10: Architecture générale de l'application .....</i>	<i>- 18 -</i>
<i>Figure-11 Séquences vidéo originales vs dégradées .....</i>	<i>- 19 -</i>
<i>Figure- 12 Structure générale du test subjectif.....</i>	<i>- 20 -</i>
<i>Figure-13. Échelle d'évaluation du test subjectif .....</i>	<i>- 20 -</i>
<i>Figure-14 Architecture Feed-forward du RNN.....</i>	<i>- 21 -</i>
<i>Figure-15 Corrélation entre les valeurs de MOS réelles et prédites de la DB d'apprenti. ....</i>	<i>- 22 -</i>
<i>Figure-16 Corrélation entre les valeurs de MOS réelles et prédites de la DB de test. ....</i>	<i>- 22 -</i>
<i>Figure-17 Corrélation entre les valeurs réelles et prédits de MOS. ....</i>	<i>- 23 -</i>
<i>Figure-18 Impact de PER et LBW sur la MOS. ....</i>	<i>- 24 -</i>
<i>Figure-19 Impact de SBR et LBW sur la MOS.....</i>	<i>- 24 -</i>
<i>Figure-20 Impact de la gigue et LBW sur la MOS. ....</i>	<i>- 25 -</i>
<i>Figure-21 Impact de la gigue et PER sur la MOS. ....</i>	<i>- 25 -</i>

# Liste des tableaux

<i>Tableau 1 : Échelle d'opinion pour le test ACR</i> .....	- 9 -
<i>Tableau 2 : Échelle d'opinion pour le test DCR</i> .....	- 10 -
<i>Tableau 3: Combinassions d'ensemble de données</i> .....	- 17 -
<i>Tableau 4: Extrait de la base des combinaisons</i> .....	- 18 -
<i>Tableau 5: Valeurs fixe des paramètres non observés</i> .....	- 23 -

# Chapitre 1

## Préambule

### 1.1 Laboratoire de recherche: LISSI

Le Laboratoire Images Signaux & Systèmes Intelligents (LISSi - EA3956) a été conjointement créé en 2005 par la Faculté des Sciences et l'Institut Universitaire de Technologies de l'Université Paris-Est Créteil Val-de-Marne (UPEC).

Le Laboratoire est impliqué dans de nombreux aspects et applications concernant le traitement du signal et de l'image, la robotique, l'intelligence artificielle, les systèmes complexes, etc. Il fournit à ses doctorants l'occasion d'acquérir de l'expérience dans divers domaines par la recherche universitaire et au travers de projets industriels.

Avec un large spectre de compétences dans les domaines des sciences et technologies de l'information, de la communication et des systèmes, le LISSI couvre les sections 7 et 8 du comité national (27, 61 et 63 du CNU). De l'information aux systèmes, de la technologie à l'humain et aux usages, les activités de recherche du LISSI concernent la modélisation et la conception de systèmes logiciels et matériels, les études en algorithmique, bioinformatique, bases de données et intelligence artificielle, le traitement et l'analyse des signaux et des images. Fort de cette diversité, le LISSI renforce son originalité en alliant théorie, outils, expérimentations et applications dans tous ses domaines de compétence scientifique, en favorisant l'émergence de projets interdisciplinaires au sein du laboratoire (image, sécurité numérique, handicap,...) et à l'interface d'autres laboratoires et domaines scientifiques (mathématiques, sciences du vivant, santé, neurosciences...).

Le LISSI regroupe aujourd'hui un peu plus de 50 personnes – dont 32 enseignants-chercheurs ou chercheurs permanents ou associés, 20 doctorants ou Post-Doc et ATER et 3 personnels administratifs et techniques (septembre 2011) – réparties sur trois sites : Vitry sur Seine, Créteil, Sénart-Fontainebleau.

### 1.2 Contextes

Ce stage s'intègre dans les cadres de la fin de la cinquième année du diplôme de Master 2 Recherche en Informatique de l'IRISA en co-habilitation avec l'université de Bretagne Occidentale, parcours systèmes et objets communicants mobiles. Il se déroule du 01/02/12 au 30/06/12.



# Chapitre 2

## Introduction

### 2.1 La Qualité d'Expérience Utilisateur

La QoS (Quality of Service) est apparu dans les années 90 pour désigner un ensemble de techniques permettant d'assurer l'acheminement du trafic réseau sensible comme la voix ou les applications. Depuis, l'acronyme QoS a été utilisé pour désigner l'amélioration de la performance réalisée par le matériel et / ou des logiciels. Mais avec le développement rapide du marché pour le streaming vidéo d'une année à l'autre, les métriques de la qualité de service (QoS) comme la bande passante, le délai, la gigue et le taux de perte qui sont généralement utilisés pour garantir les services, ne parviennent pas à mesurer la subjectivité associée à la perception humaine. Les opérateurs réseaux ont tendance à muter vers des politiques fondées sur une approche globale de la qualité de bout-en-bout, ainsi est née la qualité de l'expérience (QoE : Quality of Experience).

La QoE est une mesure du jugement personnel de l'utilisateur selon son expérience vécue, sur la qualité globale du service fourni par les opérateurs et fournisseurs de services Internet. En effet, la notion de l'expérience utilisateur a été introduite pour la première fois par le Dr Donald Norman, évoquant l'importance de la conception d'un service centré utilisateur [4]. Gulliver et Ghinea [5] décomposent la QoE en trois composantes : l'assimilation, le jugement et la satisfaction. La qualité d'assimilation est une mesure de la clarté du contenu d'un point de vue informatif. Le jugement de qualité reflète la qualité de présentation. La satisfaction indique le degré d'appréciation globale de l'utilisateur.

La Qualité de l'Expérience (QoE) est une approche globale de la qualité sur le cycle de vie du service. Elle consiste à mesurer les détails qui peuvent influencer l'ensemble d'une prestation, par exemple dans le cas du streaming vidéo : mesure de la qualité avant l'émission (propriétés du fichier source: qualité d'image, résolution,..), de la qualité d'émission (propriétés vidéo utilisées par l'encodeur, performance de la machine, protocole utilisé), de la qualité du lien (les métriques QoS: bande passante, taux de perte, délai, gigue...), de la qualité de réception (connexion internet du client, type d'écran...) et enfin le profil de l'utilisateur (âge, sexe, compétence visuelle, expérience avec le multimédia ...). La figure 1 illustre un aperçu sur la QoE dans le streaming vidéo.

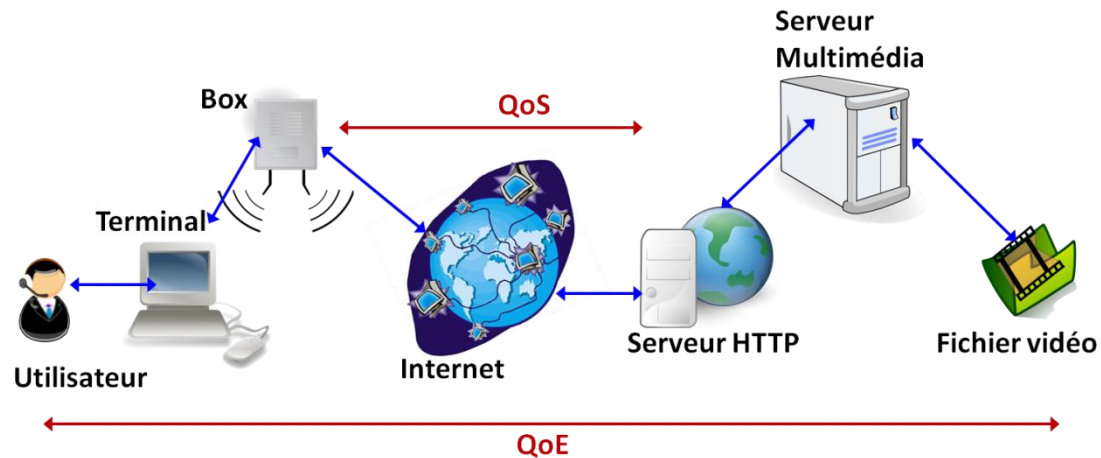


Figure-1: QoE de bout-en-bout

La QoE est basée sur une mesure subjective de la qualité, alors que la QoS est basée sur une mesure objective des éléments qui composent le service. Cependant, du fait que la QoS est parmi les facteurs qui ont un impact dur la QoE, il ne peut exister un contrôle de la QoE sans connaitre la corrélation entre les métriques de la QoS et la QoE.

## 2.2 Évaluations de la QoE

On trouve dans la littérature deux approches pour mesurer la qualité : les méthodes objectives et les évaluations subjectives. Les évaluations subjectives sont réalisées par des sujets humains qui sont invités à mesurer la qualité globale perçue dans un environnement contrôlé. La mesure la plus fréquemment utilisée est la note moyenne d'opinion (MOS : *Mean Opinion Score*), recommandée par l'union internationale de télécommunication (UIT) [6]. Plus de détails sur le MOS sont expliqués dans la section 3.2.2. Les méthodes objectives [7] qui sont basées généralement sur des algorithmes, comparent les échantillons originaux à ceux traités par les codeurs. Il existe des méthodes simples qui mesurent la qualité vidéo en faisant une différence entre les images comme le *Peak Signal to Noise Ratio* (PSNR) ou le *Mean Square Error* (MSE). Et d'autres plus complexes tels que *Normalized Video Fidelity Metric* (NVFM), ou *Moving Picture Quality Metric* (MPQM) [8]. Les mesures objectives de la qualité audio les plus connues sont : le *Perceptual Speech Quality Measures* (PSQM), le *Segmental SNR* et le rapport signal à bruit (SNR) [9, 10, 11]. Pour vérifier l'exactitude de ces méthodes, les performances des métriques objectives, doivent être corrélées avec les évaluations obtenues lors des tests subjectifs.

## 2.3 Paramètres affectant la QoE

La QoE est affectée par plusieurs facteurs de natures différentes, et l'un des objectifs de notre stage est d'évaluer l'impact de différents paramètres possibles sur la qualité. Ces paramètres peuvent être classifiés en quatre catégories :

### 2.3.1. Paramètres réseau

Les paramètres de la QoS reflètent l'état du réseau et ont impact direct sur la QoE. La dégradation de chaque paramètre réseau conduit à la dégradation de la qualité du streaming vidéo, nous citons les plus importantes :

### ***Perte de paquets***

La perte des paquets est causée par la congestion du réseau ou l'arrivée tardive des paquets. En effet, après un certain délai, le paquet est jugé comme perdu même s'il arrive dans les instants suivants. Le taux de perte est considéré comme l'un des motifs majeurs affectant la QoE.

### ***Latence (délai de bout-en-bout)***

Elle définit le temps que parcourt un paquet entre l'émission et la réception. Le délai prend en compte principalement deux phases : le temps que passe un paquet dans la file d'attente des nœuds le long du parcours et le temps de propagation du paquet sur le médium physique. Les applications temps réel comme vidéoconférence ou voix sur IP, sont très sensibles à ce facteur.

### ***Gigue***

Elle désigne la variation du délai dans le temps d'arriver des paquets à destination. Elle est généralement causée par l'utilisation de chemins différents pour atteindre la même destination, et par les différents algorithmes utilisés en file d'attente. Si la valeur de la gigue est importante cela conduit à une dégradation de la qualité des applications interactives en temps réel. Dans le streaming vidéo, les effets de la gigue peuvent être dissimulés en utilisant une mémoire-tampon du côté récepteur. Ce tampon de gigue provoque un retard détectable au début du lancement du flux. Mais, au-delà d'un certain seuil, le tampon n'aura aucun effet et une dégradation semblable à celle causé par la perte de paquet aura lieu. En effet, si les paquets arrivent en désordre et en retard par rapport au temps de maintien, ils seront ignorés au niveau du récepteur.

### ***Bande Passante***

Elle définit la mesure de la quantité de données numériques transmises par unité de temps. Avec l'accession de la télévision HD et de la vidéo 3D, les besoins en bande passante ont largement augmenté. Ce constat implique une évolution du codage source et du codage canal pour permettre une qualité d'usage satisfaisante. La bande passante a une influence significative sur la QoE si les besoins applicatifs sont importants. En effet, un flux multimédia transitant sur une largeur de bande inférieure à ses besoins, cela entraînera un temps d'attente important du côté récepteur et dégrade la qualité perçue.

## **2.3.2 Caractéristiques du contenu**

Les propriétés et caractéristiques de la vidéo ont un impact considérable sur la QoE. Ci-après quelques paramètres :

### ***Disponibilité de la vidéo***

La disponibilité d'une vidéo est le premier facteur influençant la qualité perçue du service. Des problèmes récurrents de disponibilité pour un même fournisseur aboutissent à un abandon définitif de ce fournisseur par l'utilisateur, dans un monde où tout doit être disponible en permanence et très rapidement accessible.

### **Temps d'apparition de la première image**

Idem, ce délai influence énormément la perception que l'on a du service.

### **Résolution et Frame rate (Image par seconde)**

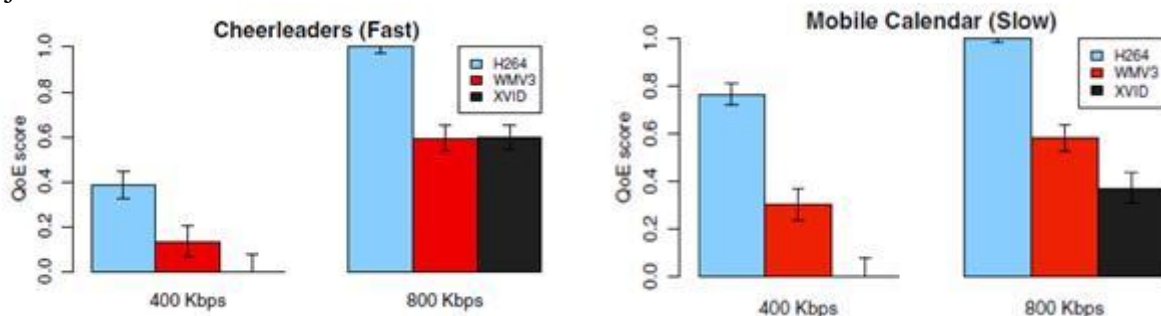
La réduction du débit requis par une application est très importante pour les applications de streaming. L'ajustement de la résolution de la vidéo ainsi que le nombre d'images par seconde peut impacter directement la QoE, du moment qu'ils jouent un rôle important dans le séquençement de la vidéo et sa qualité de visualisation.

### **Complexité du mouvement**

L'influence de la résolution et du taux d'images diffère d'une vidéo à une autre. En effet, prenons le cas d'une séquence dont la complexité du mouvement n'est pas importante, la qualité de la vidéo peut rester invariante en diminuant le taux d'images. Ainsi on peut optimiser le nombre de frames.

### **Contenu de la vidéo**

Il est à noter que le thème de la vidéo peut différencier le jugement de l'utilisateur. En effet, dans une étude menée par [12] sur l'impact du thème, une expérience a été réalisée en visualisant un match de football. La sélection des participants a été en fonction de leur intérêt pour le football. Les résultats ont montré que les fans de football ont trouvé la qualité de seulement 6 images par seconde vidéo acceptable de 80 % du temps alors que la moyenne des taux de trame couramment utilisée est de 25 images par seconde. Les auteurs concluent, que le système visuel humain peut tolérer un certain degré de distorsion et que les participants peuvent accepter une mauvaise qualité de la vidéo, s'il y a suffisamment d'intérêt pour le sujet.



**Figure-2 : QoE composé de vidéo de différent codec [13].**

### **Codec**

L'envoi d'une vidéo non compressée sur le réseau nécessite de grandes quantités de bande passante et d'espace de stockage. Par conséquent, les vidéos sont compressées avec des algorithmes de compression très efficaces. Un grand nombre de codecs vidéo sont pris en charge par les services vidéo tels que : H.262, H.263, H.264, WMV3 et WVID. La figure 2 montre que le codec impacte fortement sur la QoE.

### **Livraison de la vidéo**

Elle évalue la capacité d'une application d'assurer des flux continus et synchronisés de la vidéo, malgré les limitations du réseau sous-jacent.

### 2.3.3 Type du terminal

Avec les progrès de l'industrie des télécommunications, des services multimédias en streaming sont déployés sur différents types de terminaux: Smartphone, Ordinateur portable, Tablette, PDA et TV. Ces équipements se différencient les uns des autres par leur résolution, leur processeur et leur taille d'affichage. Ci-après quelques paramètres de l'appareil affectant la QoE.

#### *Ecran d'affichage*

Les fournisseurs de réseau peuvent évaluer la QoE de l'abonné en utilisant les paramètres de la QoS, mais aussi en tenant compte du type de terminal avec ses différentes caractéristiques. Par exemple, deux utilisateurs avec les mêmes conditions de réception peuvent avoir des qualités différentes uniquement en raison de la résolution de leur terminal.

#### *Charge du CPU*

Elle décrit les ressources de calcul utilisées par le système et le navigateur Web et affecte directement la QoE de l'application (e.g. vidéo affichée). En fait, lorsque le CPU est trop chargé, les informations reçues par le *Smartphone* ne peuvent être traitées facilement, et donc, la vidéo sera bloquée avec des images figées.

#### *Charge de la mémoire*

Elle reflète le taux d'utilisation de la mémoire qui est indispensable dans tout type de calcul.

#### *Type de navigateur*

Dans le streaming d'une vidéo, le navigateur est le logiciel souvent utilisé pour la visualisation. Comme chaque navigateur a ses performances, sa propre façon d'afficher les pages Web et sa sensibilité envers la vitesse de chargement, le type de navigateur et même la version utilisée impactent considérablement la QoE.

### 2.3.4 Facteurs psychologiques

La Qualité d'Expérience est une mesure subjective de l'adéquation d'un service par rapport à ce qu'en attend un client. Elle mesure le "rendu" de l'utilisation d'un service (navigation internet, appel téléphonique, émission TV, etc.). Les systèmes de mesures de la Qualité d'Expérience essaient de mesurer les facteurs influant sur la qualité perçue par l'utilisateur (par exemple: le temps de zapping TV, les dégradations sur l'image). La QoE est en partie liée avec la Qualité de service, qui mesure des paramètres techniques et objectifs. La QoS est cependant rarement mesurée chez l'utilisateur, et est orientée vers le moyen par opposition à la QoE qui est orientée vers l'utilisateur.

L'estimation des variations du niveau de qualité d'une personne à l'autre dépend de plusieurs facteurs qui sont liés à la personne, ses préférences et son environnement.

Nous pouvons préciser certains de ces paramètres :

- Caractéristique de l'utilisateur (connaissances, expérience, langue, âge, sexe, ...)
- Comportement de l'utilisateur (degré d'intérêt pour le contenu)
- Dans le cas d'un test QoE subjectif, familiarité avec la tâche (expert ou non-expert)

# Chapitre 3

## Travaux du stage

Cette partie présente les travaux réalisés durant le stage. Des réflexions et idées aux expériences réalisées. Elle expose aussi des analyses sur les résultats obtenus et les intentions de travaux futurs.

### 3.1. Objectifs

Fort de l'ensemble des équipements à disposition, l'objectif du stage est de mener une étude sur les facteurs qui ont un impact sur la QoE ; Concevoir une plateforme afin de mener des expériences pour l'évaluation subjective de la qualité vidéo; Générer une base de données avec les résultats récoltés; Et de développer un modèle pour la prédiction de QoE en fonction des paramètres de QoS.

### 3.2 .Méthodologie

La qualité du streaming vidéo est affectée par de nombreux facteurs comme le débit, les pertes de paquets, le type de codage, la gigue, etc. cependant, la qualité perçue n'est pas linéairement corrélative à la variation d'aucun de ces paramètres [3]. En effet, le problème de la détermination de la qualité est complexe, et il n'est pas possible de le résoudre par un modèle mathématique qui regroupe tous ces paramètres. Les travaux menés par [3] ont montré que les réseaux de neurones aléatoires (RNN) sont des outils appropriés pour résoudre le problème de la prédiction de la qualité audio en fonction des paramètres de QoS. Nous illustrons notre travail en mettant en œuvre un RNN pour intercepter la relation non linéaire entre des mesures non subjectives (paramètres réseau et de l'encodage) et la qualité du streaming vidéo perçue par des sujets d'humains.

Avant tout, nous avons déterminé un ensemble de facteurs qui affecte la perception de la qualité. De ce fait, nous avons choisi les plus importants paramètres vidéo et réseau qui assure la transmission, pour chaque paramètre nous avons choisi trois valeurs couvrant la gamme des valeurs possibles. Ensuite, nous avons produit une base de données composée des échantillons correspondant à différentes combinaisons des paramètres choisis. Par la suite, un test subjectif sur la qualité du streaming vidéo a été effectué. En résumé, des échantillons de flux vidéo ont été envoyés de la source à la destination tout en contrôlant le réseau de transmission. Et des individus ont été invités à évaluer la qualité des échantillons. Il existe plusieurs méthodes subjectives dans les recommandations de l'UIT-R[15] ou de l'UIT-T[6,14]. Ces méthodes sont présentées dans la section suivante. En se basant sur les scores donnés par les sujets, des filtrages statistiques ont été réalisés pour éliminer les notes des sujets suspectés d'attribuer des scores incertains (voir section 3.2.2). L'étape qui a suivie est le calcul des valeurs de MOS pour chacune des combinaisons en suivant les recommandations de l'UIT-T[6]. Ensuite, nous avons généré une base de données contenant les valeurs de MOS et les valeurs de paramètres correspondants. Enfin, une architecture appropriée de Réseau de Neurones Aléatoire a été

choisie. La base de données construite précédemment a été divisée en deux parties : une pour entraîner le RN et l'autre pour vérifier son exactitude.

### 3.2.1 Mesure subjective de la qualité vidéo

L'Union Internationale des Télécommunications (UIT) et le Groupe des Experts de la Qualité Vidéo (VQEG : Video Quality Experts Group) ont défini les méthodologies à suivre et les conditions à respecter lors des tests subjectifs. Ces recommandations se résument en trois phases : la sélection d'un nombre minimal de visionneurs (sujets), puis la diffusion des séquences vidéo en question dans des conditions contrôlées. Après la fin de chaque séquence, les visionneurs sont invités à évaluer la séquence vidéo en attribuant une note. Les méthodes de tests subjectifs sont décrits dans de l'UIT-R T.500-11 (2002) [15] et de l'UIT-T Rec. P.910 (1999) [14]. Ces méthodes offrent des conseils sur les de conditions d'observation, la référence pour les observateurs, la sélection du matériel de test, la procédure d'évaluation et les méthodes d'analyse statistique.

Les méthodes subjectives les plus connus sont:

#### 3.2.1.1 Échelle de dépréciation à double stimulus (DSIS)

La méthode DSIS (Double Stimulus Impairment Scale), consiste à présenter au sujet des paires de séquences vidéo, une séquence dégradée et l'autre de référence. La séquence dégradée est toujours affichée en présence d'une séquence de référence. Les sujets donnent un score selon une échelle de dépréciation, « imperceptible, sensible mais pas ennuyeux, un peu ennuyeux, ennuyeux, et très ennuyeux ». Cette échelle est peut également noter sur 5 points. Le score 5 représente imperceptible et le score 1 très ennuyeux.

#### 3.2.1.2 Méthode de stimulus unique

Contrairement à la méthode précédente, les sujets regardent plusieurs scènes séparées. Cette méthode peut se dérouler selon deux façons: la première est le stimulus unique sans répétition, dans laquelle les sujets regardent une scène de test une seule fois sans la possibilité de la répéter ; et la seconde est le stimulus unique avec répétition, où il est possible de répéter plusieurs fois les scènes de tests. De plus, trois différents types de méthodes de notation sont utilisés:

##### *Adjective*

C'est la même que celle décrite dans l'échelle de dépréciation à double stimulus avec la différence que les échelles de moitié sont autorisées.

##### *Numérique*

Dans laquelle, une échelle de 11 notes numériques est utilisée. Ceci est utile si une référence n'est pas disponible.

##### *Non catégorique*

Dans celle-ci, une échelle continue sans numérotation est utilisée. Alternativement une large gamme de numéros peut être utilisée, par exemple, de 0 à 100.

#### 3.2.1.3 Méthode de comparaison de stimulus

Cette méthode est utilisée lorsque deux moniteurs bien assortis sont disponibles. Les différences entre les paires de scènes sont notées dans l'une des deux façons suivantes:

**Adjective**

Il s'agit d'une échelle de 7 catégories de 3 à -3: L'échelle est labellisée par: « beaucoup mieux, mieux, un peu mieux, même, légèrement pire, pire, et bien pire ».

**Non catégorique**

Celle-ci est très similaire à la précédente, où une échelle continue est utilisée sans ou avec numérotation.

**3.2.1.4 Évaluation de la qualité continue en stimulus unique (SSCQE)**

Dans la méthode SSCQE (Single Stimulus Continuous Quality Evaluation), les sujets regardent une séquence vidéo de 20 à 30 minutes sans aucune référence à la séquence originale. Les sujets attribuent des notes à chaque instant où ils constatent un changement de qualité, en utilisant un curseur de permanence de la qualité perçue sur une échelle de «médiocre» à «excellente». Cela correspond à une échelle numérique de 0 à 100.

**3.2.1.5 Échelle continue de la qualité en double stimulus (DSCQS)**

La DSCQS (Double Stimulus Continuous Quality Scale) est utile quand on est dans l'impossibilité de fournir des conditions de tests qui montrent la gamme complète de la qualité. De même que le DSIS, les séquences vidéo sont présentés en paires : l'originale et la dégradée. Mais à la différence du DSIS, les sujets sont invités à évaluer la qualité des deux séquences et non pas uniquement la séquence dégradée, en ignorant qu'elle est la séquence originale entre les deux. Les séquences vidéo sont de très courtes (soit 10 secondes) et la qualité est mesurée sur une échelle continue allant de «médiocre» à «excellente». Ceci est équivalent à une échelle numérique de 0 à 100.

Les méthodes mentionnées ci-dessus sont décrits en détail dans la recommandation UIT-R. BT.500-10 [15] et sont principalement destinés aux signaux de télévision. Par ailleurs, d'autres méthodes basées sur de légères modifications et adaptations de ces méthodes ont été décrites dans les recommandations de l'UIT-T Rec. P.911 [14], pour l'évaluation subjective des services multimédias. Paris eux, la Note par Catégories Absolues (ACR : Absolute Category Rating) et la Note par Catégories de Dégradation (DCR : Degradation Category Rating). La Note par Catégories Absolues (ACR) est la méthode la plus couramment utilisée et elle emploie la note moyenne d'opinion (MOS) comme un indicateur de mesure. La Note par catégories de dégradation (DCR) est également utilisée dans certaines situations et elle emploie la note de dégradation moyenne d'opinion (DMOS) comme métrique.

**3.2.1.6 Note absolue catégorie (ACR)**

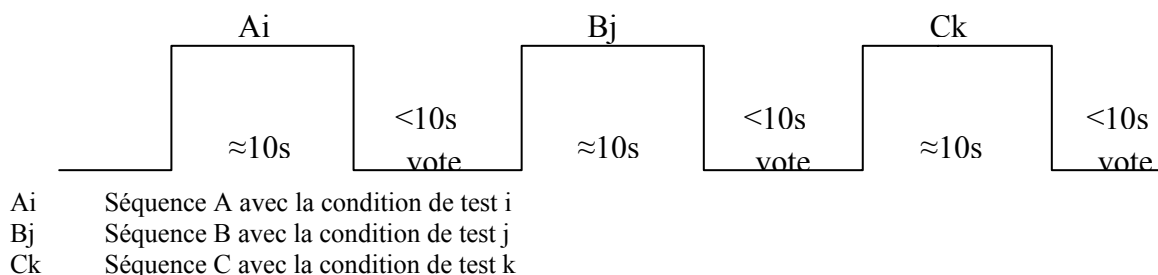
Dans la méthode ACR, des séquences vidéo sont présentées une par une aux sujets sans regarder la vidéo de référence. Après la fin de chaque séquence, ils sont invités à donner une évaluation de la qualité. La note de qualité est basée sur une échelle d'opinion, comme indiquée dans le tableau 1.

**Tableau 1 : Échelle d'opinion pour le test ACR**

Catégorie	Qualité vidéo
5	Excellente
4	Bonne
3	Assez bonne
2	Pauvre
1	Médiocre



La note moyenne d'opinion (MOS) est alors calculée comme une moyenne de tous les scores d'opinion des sujets.



**Figure-3** : Présentation du stimulus dans la méthode ACR

La figure-3, montre l'échelonnement de la présentation vidéo dans le temps. La période du vote est égale ou inférieure à 10 secondes.

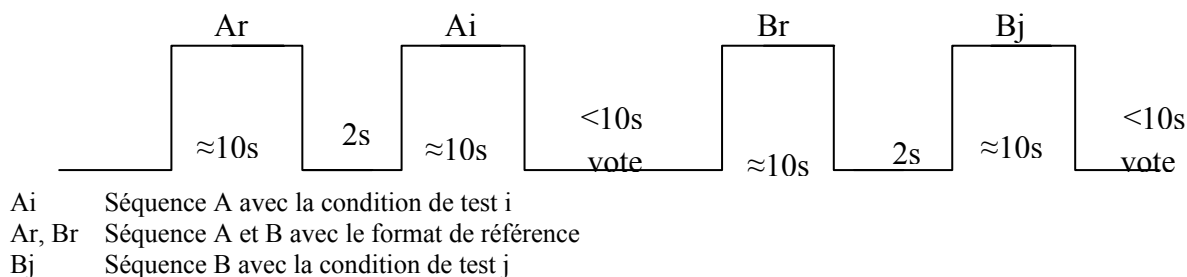
### 3.2.1.7 Note par catégories de dégradation (DCR)

La méthode DCR (Dégradation Category Rating) est plus efficace lorsque la différence de qualité dans une séquence vidéo est minimale. Avec la méthode ACR les sujets ont du mal différencié les dégradations de la qualité par exemple entre le score 3 et 4. Dans cette méthode, les séquences vidéo sont présentées en paires. La vidéo originale est toujours présentée dans le premier échantillon de chaque paire, c'est-à-dire, la vidéo originale sans aucune dégradation. La seconde séquence est la même vidéo mais modifiée par les conditions de tests. Dans cette méthode, les utilisateurs sont invités à évaluer la déformation de la deuxième séquence par rapport à l'originale. Les échelles de notation ou les niveaux de dégradation sont présentés dans le tableau 2.

**Tableau 2** : Échelle d'opinion pour le test DCR

Catégorie	Qualité vidéo
5	Imperceptible
4	Perceptible, mais pas ennuyeux
3	Un peu ennuyeux
2	Ennuyeux
1	Très ennuyeux

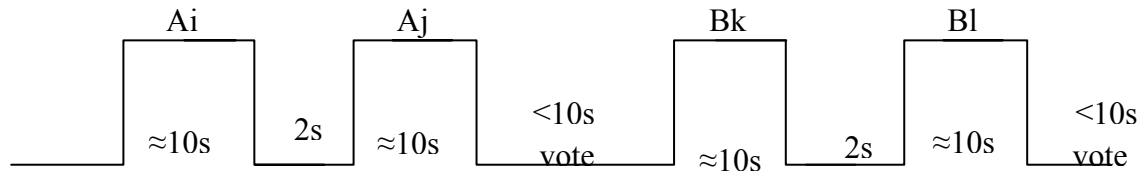
La figure 4 montre l'échelonnement dans le temps pour la présentation des clips vidéo. Le temps des séquences doit être inférieur ou égal à 10 secondes.



**Figure-4** : Présentation du stimulus dans la méthode DCR

### 3.2.1.8 Méthode de comparaison par paires (PC)

Dans la méthode, des comparaisons par paires de séquences de tests sont menés à plusieurs reprises. En comparaison à d'autres méthodes telles que le stimulus unique et le stimulus double, les séquences de tests sont combinés dans toutes les combinaisons possibles. En outre, toutes les paires de séquences sont présentées dans les deux ordres, les sujets n'ont seulement à fournir une préférence entre les séquences de chaque paire au lieu d'attribuer un score. Le modèle de temps pour la présentation des séquences vidéo est présenté dans la figure-5. Comme précédemment, la durée des votes devrait être inférieure ou égale à 10 s.



Ai, Aj Séquence A avec la condition de test i et j respectivement  
 Bk, Bl Séquence A avec la condition de test i et j respectivement

Figure-5 : Présentation du stimulus dans la méthode PC

### 3.2.2 Analyse statistique de la MOS

Désignons par  $S = \{\sigma_1, \sigma_2, \sigma_s\}$  l'ensemble des échantillons, et par  $N$  le nombre de sujets humains dans effectuant les tests et par  $u_{is}$  l'évaluation subjective de l'échantillon  $\sigma_s$  faite par l'individu  $i$ . L'ensemble des valeurs  $(u_{is})_{i=1, \dots, N}$  contiendraient sans doute des variations dues aux différences dans le jugement de chaque individu. En effet, il se peut que quelques utilisateurs négligent l'importance du test ou ne s'appliquent pas lors de l'évaluation. Ce qui mènera à des données imprécises pour la phase d'apprentissage. C'est pourquoi, il est nécessaire de faire un filtrage statistique sur l'ensemble de données brutes. La recommandation UIT-R BT.500-10 [15] est une des références la plus répandue. Elle décrit en outre, un procédé explicite pour enlever les évaluations des sujets qui ont conduit à des notes non conformes.

Dénotant par  $\bar{u}_s$  la moyenne des évaluations de l'échantillon  $\sigma_s$  sur l'ensemble des sujets.

$$\bar{u}_s = \frac{1}{N} \sum_{i=1}^N u_{is}$$

Soit  $[\bar{u}_s - \Delta_s, \bar{u}_s + \Delta_s]$  l'intervalle de confiance à 95% obtenu à partir de  $(u_{is})$ .

Où  $\Delta_s = 1,96 \delta_s / \sqrt{N}$ , et  $\delta_s = \sqrt{\sum_{i=1}^N \frac{(u_{is} - \bar{u}_s)^2}{N-1}}$

La recommandation [15] mentionne qu'on doit vérifier si la distribution des notes est normale en calculant le coefficient de *kurtosis* pour le test  $\beta_2$ . Si  $\beta_2$  est entre 2 et 4, la distribution peut être supposée normale. Dénotons par  $\beta_{2s} = m_{4s} / m_{2s}^2$  où

$$m_{xs} = \frac{1}{N} \sum_{i=1}^N (u_{is} - \bar{u}_s)^x$$

Si  $2 \leq \beta_{2s} \leq 4$  alors on peut considérer que la distribution  $(u_{is})_{i=1, \dots, N}$  est normale. Pour chaque individu  $i$ , nous calculons deux valeurs de nombre entier  $L_i$  et  $R_i$ , comme suit :

soit  $L_i = 0$  et  $R_i = 0$

pour chaque échantillon  $\sigma_s \in S = \{\sigma_1, \dots, \sigma_s\}$

si  $2 \leq \beta_{2s} \leq 4$ , alors

si  $u_{is} \geq \bar{u}_s + 2\delta_s$  alors  $R_i = R_i + 1$   
 si  $u_{is} \leq \bar{u}_s - 2\delta_s$  alors  $L_i = L_i + 1$

sinon

si  $u_{is} \geq \bar{u}_s + \sqrt{20}\delta_s$  alors  $R_i = R_i + 1$   
 si  $u_{is} \leq \bar{u}_s - \sqrt{20}\delta_s$  alors  $L_i = L_i + 1$

Enfin, si  $(L_i + R_i)/S > 0,05$  et  $|(L_i - R_i) / (L_i + R_i)| < 0,3$  alors les notes de l'individu  $i$  doivent être supprimées.

Après le filtrage des notes et l'élimination des évaluations qui pourraient conduire à des estimations incohérentes, notre BD de MOS ainsi construite pourra être employé pour l'entraînement et la validation.

### 3.2.3 Les réseaux de neurones aléatoires

Les réseaux de neurones aléatoires (RNN) sont une variante du concept de réseaux de neurones dont le fonctionnement est plus proche du modèle biologique.

Les neurones dans un RNN, ne s'activent pas de manière continue, mais applique la notion d'échange de signaux au cours du temps. Ces signaux peuvent être excitateurs s'ils font augmenter le potentiel du neurone qui les reçoit, ou inhibiteurs s'ils font diminuer ce potentiel. Le neurone pourra émettre des signaux, que si son potentiel est positif.

#### 3.2.3.1 Le neurone élémentaire dans RNN

Un neurone aléatoire est caractérisé par son état interne appelé potentiel. Le potentiel est noté  $k_j(t)$  pour le neurone  $j$  à l'instant  $t$ . La valeur de  $k_j(t)$ :

- augmente d'une unité, si le neurone reçoit un signal excitateur;
- diminue d'une unité, si le neurone reçoit un signal inhibiteur;
- diminue d'une unité, si le neurone émet un signal.

Le comportement du neurone en sortie est dépendant de son potentiel :

- si  $k_j(t) > 0$ , le neurone est alors excité. Il émet donc des signaux avec une fréquence  $r_j$  ;
- sinon, il n'émet aucun signal (neurone pas excité).

Les calculs dans un RNN sont opérés en termes de probabilité d'excitation des neurones.

On note  $q_j(t)$  la probabilité que neurone  $j$  soit excité au temps  $t$  :  $q_j(t) = P[k_j(t) > 0]$ . Le neurone transforme les fréquences d'entrées en une quantité  $k_j$ .

Les distributions stationnaires de probabilités associées à ce modèle sont:

$$q_j = \lim_{t \rightarrow \infty} q_j(t)$$

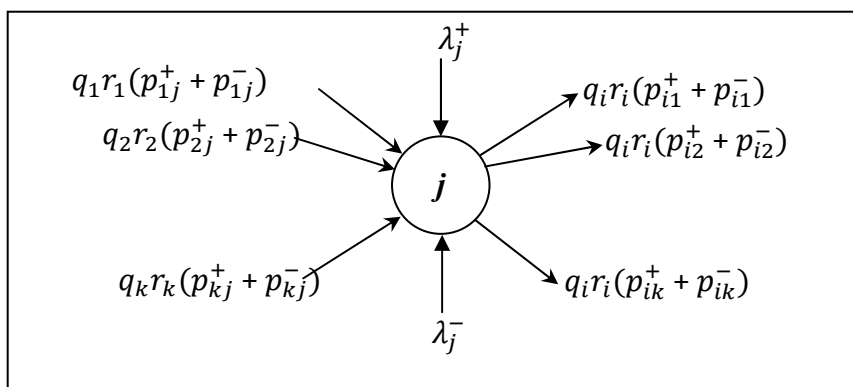


Figure-6: Représentation du neurone de RNN

- $\lambda_j^+$  taux d'arrivée du signal exogène excitateur;
- $\lambda_j^-$  taux d'arrivée du signal exogène inhibiteur;
- $q_k r_k (p_{kj}^+ + p_{kj}^-)$  taux d'arrivée des signaux excitateurs et inhibiteurs au neurone;
- $q_j r_j (p_{jk}^+ + p_{jk}^-)$  taux d'émission des signaux excitateurs et inhibiteurs par neurone.
- $r_j$  taux d'émission de signaux par le neurone.

Le potentiel du neurone  $j$  est calculé ainsi :

$$q_j = \frac{\lambda_j^+}{\lambda_j^- + r_j}$$

### 3.2.3.2 Fonctionnement en réseau

Les neurones aléatoires une fois assemblés en réseau, reçoivent des signaux parvenant de l'extérieur. Ces signaux constituent les entrées du réseau de neurones aléatoires. Chaque neurone d'entrées  $j$  reçoit des signaux excitateurs ou inhibiteurs, suivant une loi de poisson au taux de  $\Lambda_j$  pour les signaux extracteurs ou  $\lambda_j$  pour les signaux extracteurs ou inhibiteurs. De plus quand un neurone  $j$  émet un signal, ce dernier peut quitter le réseau avec probabilité  $d_j$ . Par ailleurs, il existe aussi des signaux circulant entre les neurones. Par exemple, lorsqu'un neurone  $i$  est excité et qu'il émet un signal au neurone  $j$ , le signal parvient à  $j$  avec une probabilité  $p_{ij}^+$  dans le cas d'un signal excitateur, ou  $p_{ij}^-$  dans le cas d'un signal inhibiteur ; L'ensemble des signaux parvenant à un neurone  $j$  de la part de  $k$  neurones, est déterminé par :

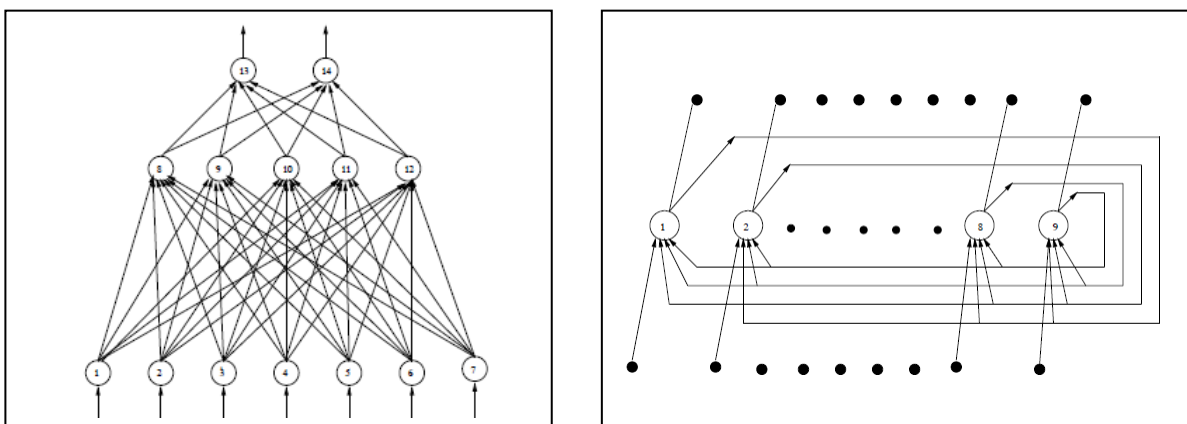
$$\lambda_j^+ = \Lambda_j + \sum_{i=1}^k q_i r_i p_{ij}^+$$

$$\lambda_j^- = \lambda_j + \sum_{i=1}^k q_i r_i p_{ij}^-$$

Dans les RNN, comme dans les réseaux de neurones classiques, on parle aussi de poids des connexions qui déterminent le comportement du réseau. Mais les RNN, utilisent deux poids par connexion  $w_{ij}^+$  et  $w_{ij}^-$  toujours positifs ou nuls:

$$w_{ij}^+ = r_i p_{ij}^+$$

$$w_{ij}^- = r_i p_{ij}^-$$



(a) Architecture Feed-forward

(b) Architecture récurrente

**Figure-7** : Architecture des RNN

### 3.2.3.3 Architecture des RNN

Les réseaux de neurones aléatoires adoptent les mêmes architectures des réseaux de neurones classiques :

- architecture *Feed-forward*, à une ou plusieurs couches ;
- architecture récurrente.

### 3.2.3.4 L'apprentissage des RNN

L'apprentissage d'un réseau de neurones aléatoires permet comme dans le cas d'un réseau de neurones classique, de fixer les paramètres de la fonction réalisée par le RNN afin d'obtenir le comportement souhaité. Dans un RNN les poids des connexions se divisent en deux matrices  $W^+$  et  $W^-$  qui représentent respectivement les poids des connexions excitatrices et inhibitrices du réseau.

Il existe plusieurs algorithmes utilisés pour l'apprentissage de réseaux de neurones aléatoires : l'algorithme de rétropropagation [16, 17], l'algorithme RPROP [18], l'apprentissage par renforcement [19]

Dans ce qui suit, nous présentons l'algorithme de rétropropagation qui est le plus commun en matière de réseaux de neurones. Nous allons tout d'abord, le présenter dans le contexte des réseaux de neurones classiques. Ensuite, nous verrons les améliorations qui ont été apportées par [16, 17] pour l'adapter à un RNN.

#### Algorithme de rétropropagation [20]

Le principe est le suivant : soit un réseau de neurones possédant  $n$  entrées et  $d$  sorties et des couches cachées. Pour chaque neurone de sortie, on cherche à connaître la contribution des poids de ce neurone à l'erreur obtenue en sortie du neurone. Une fois cette contribution déterminée, on réitère l'opération pour tous les neurones connectés en entrée de ce neurone. On fait un balayage à partir des neurones de sortie vers les entrées du réseau, en propageant les erreurs calculées à chaque couche. L'erreur calculée pour un neurone  $j$  est :

- Pour un neurone de sortie :

$$\delta_j = (y_j - u_j)y_j(1 - y_j)$$

- Pour un neurone caché :

$$\delta_j = y_j(1 - y_j) \sum_{k \in dest(j)} \delta_k \omega_{jk}$$

où  $dest(j)$  est l'ensemble des neurones connectés à la sortie de  $j$ .

La mise à jour des poids est:

$$\Delta \omega_{ij} = \alpha \delta_j y_i$$

L'algorithme obtenu pour l'apprentissage est le suivant :

```

Tant que (l'algorithme n'a pas convergé) faire
  choisir un exemple x (aléatoirement ou non)
  Pour chaque couche en partant de la couche de sortie faire
    Pour chaque neurone j de la couche faire
      calculer  $\delta_j$ 
      Pour chaque connexion  $w_{ij}$  menant à j faire
         $\Delta w_{ij} = \alpha \delta_j y_i$ 

      Fin Pour
    Fin Pour
  Fin Pour
  Pour chaque connexion  $w_{ij}$  faire
     $w_{ij} = w_{ij} + \Delta w_{ij}$  .
  Fin Pour
Fait

```

*Algorithme 1: L'algorithme de rétro-propagation du gradient de l'erreur*

### L'algorithme de rétropropagation du gradient de l'erreur [16, 17]

Est une adaptation de l'algorithme 1 utilisé pour les réseaux de neurones, qui consiste grâce à une technique de descente de gradient à mettre à jour les poids  $W^+$  et  $W^-$  en fonction de leur participation à l'erreur globale. L'erreur que l'on cherche à minimiser est :

$$E = \frac{1}{2} \sum_{i=1}^n (q_i - y_i)^2$$

où  $y = (y_1, \dots, y_n)$  est la sortie désirée. La règle employée pour modifier les poids à la  $k^e$  itération:

$$w_{ij}^+(k) = w_{ij}^+(k) - \mu \sum_{i=1}^n (q_i(k) - y_i(k)) \left[ \frac{\partial q_i}{\partial w_{ij}^+} \right] (k)$$

$$w_{ij}^-(k) = w_{ij}^-(k) - \mu \sum_{i=1}^n (q_i(k) - y_i(k)) \left[ \frac{\partial q_i}{\partial w_{ij}^-} \right] (k)$$

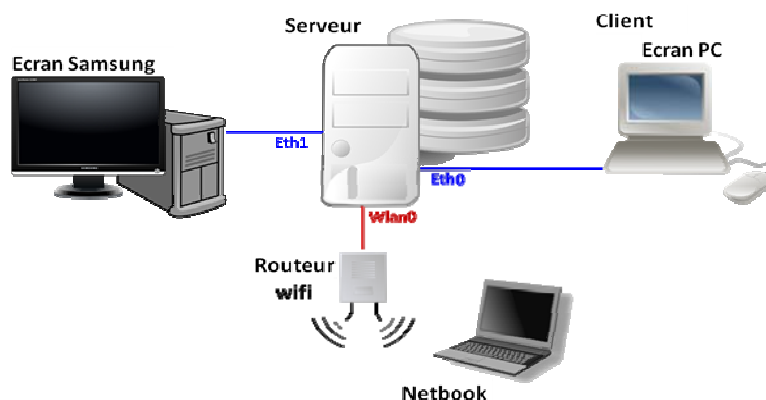
## 3.3. Expérience

Afin d'analyser l'impact des différents paramètres vidéo, réseau et applicatifs sur la qualité du streaming vidéo perçue par l'utilisateur, nous avons conçu une expérience dans laquelle des sujets humains sont invités à évaluer la qualité vidéo sur différents types d'écrans.

### 3.3.1. Configuration de l'expérience

Le dispositif expérimental est constitué d'un serveur de streaming vidéo, d'un routeur wifi et de trois clients vidéo. Trois écrans d'affichage ont été sélectionnés pour lire la vidéo du côté client. Il s'agit d'un écran *HD TV SAMSUNG "27 pouces"*, (1920x1080 pixels), d'un écran *PC DELL "17 pouces"*, (1280x1024 pixels) et d'un *Netbook ASUS "10.1 pouces"*, (1024x600 pixels). Le serveur dispose de trois cartes réseau, deux de type Ethernet destinées aux postes fixe et une carte sans fil pour le Netbook. Un émulateur réseau (*NETEM*) fait

varier les paramètres réseau au niveau des trois interfaces réseau du serveur, ainsi le trafic entre le serveur et les clients sera dégradé. Un schéma conceptuel est montré dans la figure suivante.



*Figure-8: Configuration de l'expérience*

### 3.3.2 Choix des séquences vidéo

Trois séquences vidéo ont été choisies : Vidéo1 (peu de mouvement), Vidéo2 (mouvement moyen) et Vidéo3 (mouvement rapide). Les séquences vidéo ont été intentionnellement découpées du même fichier source, afin qu'ils disposent des mêmes propriétés comme la résolution le taux d'images par seconde et le type d'encodage. En outre, les séquences sont extraites du film de science-fiction «Transformers3 ». Chaque vidéo contient 300 images, est encodée à 30 images par seconde, avec une durée de 10 secondes et une résolution de  $600 \times 300$  pixels. Un aperçu sur les séquences vidéo est présenté dans la figure 9.



*Fig 9.1 Vidéo1*

*Fig 9.2 Vidéo2*

*Fig 9.3 Vidéo3*

*Figure-9 Séquences vidéo choisies*

### 3.3.3 Choix des paramètres

En raison de la contrainte du nombre de bénévoles qui participeront à l'expérience, nous ne pouvons pas tenir compte de tous les paramètres déjà cités. Donc, nous avons sélectionné certains d'entre eux que nous avons qualifiés importants dans le jugement de la qualité et qui ont déjà prouvé séparément qu'ils ont une influence sur la perception de l'utilisateur par des travaux antérieurs. En outre, des combinaisons de paramètres associés à l'encodeur, au réseau et au type de contenu ont été choisies. Les paramètres associés à l'encodeur sont le taux d'images par seconde (FR : Frame Rate), et le débit de flux sortant de l'encodeur (SBR : Bitrate Sender). Les paramètres réseau choisis sont bande passante (LBW : Link Bandwidth), le taux de perte (PER : Packet Error Rate) et la gigue (Jitter) qui représente la variation dans

le délai. Nous n'avons pas pris en considération le délai dans nos expériences. Le délai est plus important dans les applications vocales. En effet, les applications vocales sont en temps réel et un retard dans la voix, donne à l'utilisateur l'impression que la communication est perdue. Cependant, pour le streaming vidéo, le retard n'est pas aussi important comme il a été prouvé dans [1]. De plus, nous avons effectué plusieurs expériences avant le choix définitif des paramètres où nous avons fait varier le délai jusqu'à 240 millisecondes. Les sujets étaient indifférents devant ce facteur. Pareillement, le paramètre de la gigue peut ne pas être pris en compte dans ce type d'application, car son influence sur la qualité peut être évitée en utilisant une mémoire cache. Mais nous avons estimé qu'il été important de l'introduire afin d'étudier son impact sur la qualité.

Enfin, un paramètre sur la complexité du mouvement des séquences est pris en compte. Les séquences vidéo ainsi que les paramètres choisis combinaison sont donnés dans le tableau 3.

- Le débit (SBR) : c'est le débit de flux sortant de l'encodeur. Nous avons choisi trois valeurs : 600, 800 et 1000 koctets/s.
- Le taux d'images par seconde (FR) : ce paramètre prend l'une des 3 valeurs 10, 15 et 30. Ceci est fait dans l'encodeur en laissant tomber des images uniformément.
- La bande passante (LBW) : la largeur de bande prend les valeurs de 1, 1.5 et 2 mbps.
- Le taux de perte de paquet (LR) : ce paramètre prend trois valeurs 0, 5 et 10%.
- la variation dans le délai (Jitter) : la gigue prend les valeurs de 0, 5 et 10 ms.
- Type de contenu vidéo : ce paramètre prend trois valeurs selon la Complexité du mouvement de la vidéo : Peu de Mouvement, Mouvement Moyen et Mouvement Rapide.
- Type d'écran : il s'agit du matériel sur le quel va jouer la vidéo.

Ecran	Séquences	SBR(kbps)	FR(fps)	LWB(mbps)	PER(%)	Jitter(ms)
Ecran Samsung, Ecran de PC, Netbook	vidéo1(P.M), vidéo2(M.M), vidéo3(M.R)	600, 800, 1000	10, 15, 30	1, 1.5, 2	0, 5, 10	0, 5, 10

**Tableau 3:** *Combinassions d'ensemble de données*

### 3.3.4 Génération d'ensemble de données

Si nous considérons toutes les combinaisons de ces valeurs de paramètres, nous aurons à tenir compte de  $3^7 = 2187$  combinaisons différentes. Ceci nécessiterait un travail énorme pour évaluer leurs qualités. Cependant, avec les algorithmes d'optimisation telle que les réseaux de neurones, n'ont pas besoin de toutes ces combinaisons pour apprendre le problème. Le rôle d'un réseau de neurones par exemple, est d'interpoler la qualité des parties manquantes de cet espace d'entrée potentiellement grand. Par conséquent, nous avons divisé le problème des combinaisons en deux. Tout d'abord, nous avons combiné les valeurs de la QoS à ceux du type de contenu (81 combinaisons), ensuite nous les avons concaténé avec les combinaisons liées à l'encodeur (9 répétitions de 9 combinaisons). Ceci a mené à 81 combinaisons comme montrées dans le tableau A1 en annexe. Ainsi, un ensemble de données contenant les différentes combinaisons du test subjectif est généré dans notre BD. Enfin, nous avons fait passer les 81 échantillons sur les trois types d'écrans. Le tableau 4 montre un extrait de la base des combinaisons.



Tableau 4 Extrait de la base des combinaisons

N°échant	Vidéo	SBR	FR	LBW	JITTER	PER
1	1	1000	30	2mbit	0ms	0
2	1	1000	15	2mbit	0ms	5
3	1	1000	10	2mbit	0ms	10
...						
79	3	600	30	1mbit	10ms	0
80	3	600	15	1mbit	10ms	5
81	3	600	10	1mbit	10ms	10

### 3.3.5 Application

Les 81 échantillons étant définis, nous avons développé une application web en langage script *PHP* afin de :

- 1- Contrôler les propriétés réseau entre le client et le serveur
- 2- Envoyer un flux vidéo depuis le serveur envers le client
- 3- Interagir avec le serveur via une interface client pour lancer la vidéo, afficher le flux vidéo reçue, et noter la qualité
- 4- Enregistrer les paramètres réseau, les propriétés vidéo et la note de la qualité perçue dans une base de données.

L'application interagit via le Shell de *Linux* avec deux outils, le premier est utilisé pour l'envoi du flux vidéo sur le réseau et l'autre pour le contrôle des paramètres réseau. Pour envoyer des flux vidéo nous avons employé l'outil *VLC*, qui encode des flux vidéo temps réel. Par ailleurs, l'émulateur réseau *NetEm* a été utilisé pour émuler les propriétés réseau, comme par exemple introduire un retard constant sur les paquets, ajouter un taux de perte ou introduire une gigue. Le protocole *UDP* utilisé traditionnellement pour le streaming vidéo a été employé pour acheminer les paquets entre le serveur et les clients. La figure 10 montre l'architecture générale de notre application.

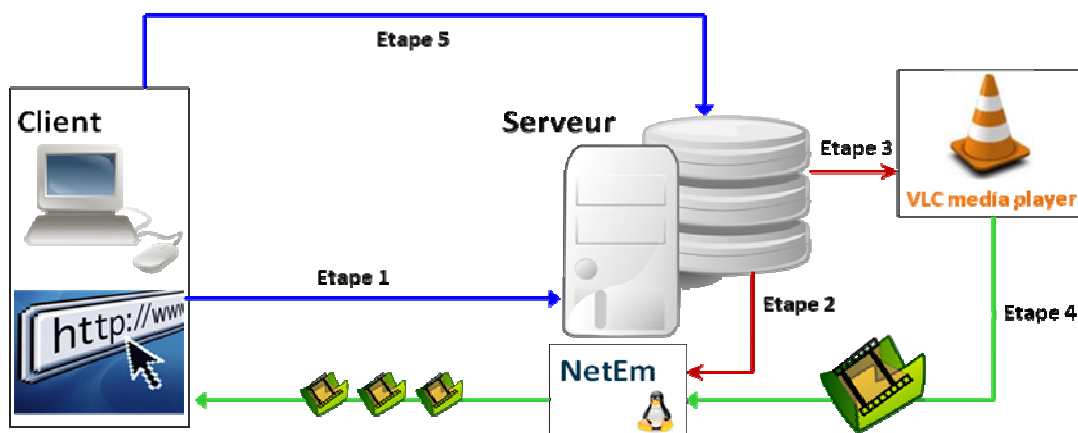


Figure-10: Architecture générale de l'application

Grâce au navigateur web, le client envoie une requête au serveur lui indiquant qu'il est prêt à recevoir une vidéo. Tout d'abord, une combinaison de paramètres réseau et vidéo est sélectionnée aléatoirement parmi les combinaisons restantes dans la base de données, et qui n'ont pas été testé avec le même client. Ensuite, l'émulateur réseau *NetEM* est exécuté avec les paramètres réseau de la combinaison sélectionnée, ainsi l'interface réseau du serveur prend les nouvelles valeurs (étape 2). La troisième étape, consiste au lancement de *VLC* qui prend quant à lui les valeurs des propriétés vidéo de la même combinaison. Le flux vidéo diffusé par *VLC*, transit par l'interface réseau contrôlé par *NetEM* avant de s'afficher sur le navigateur web du client en mode plein écran (étape 4). Enfin, après la fin de chaque séquence vidéo, l'utilisateur donne un score sur la qualité de la vidéo. Cette note sera enregistrée dans la base de données accompagnée de la combinaison des propriétés réseau et vidéo sélectionnée. Ce processus est répété jusqu'à ce que l'utilisateur termine toutes les 81 combinaisons existant dans la base.



Fig 11.1 Video1 (originale)

Fig 11.3 Video2 (originale)

Fig 11.5 Video3



Fig 11.2 Video1 (dégradé)

Fig 11.4 Video2 (dégradé)

Fig 11.6 Video3 (dégradé)

**Figure-11** Séquences vidéo originales vs dégradées

### 3.3.6 Le scénario expérimental du test subjectif

Une fois le dispositif expérimental installé, plusieurs aspects ont été pris en compte. En outre, un scénario expérimental a été étudié afin d'offrir des conditions identiques à tous les sujets. Un descriptif du scénario est présenté dans les paragraphes suivants.

#### **Les conditions d'observation**

Les conditions d'observation ont été uniformes pour tous les sujets, par exemple : les séquences de tests, le matériel de présentation, la position assise, etc

#### **Réplifications**

La réplification des séquences vidéo est requise par l'UIT-T P.911[14]. Les mêmes séquences vidéo avec les mêmes paramètres sélectionnés ont été montrées aux sujets. Ceci est important pour validation des résultats.

### Ordre de présentation

L'ordre de présentation des séquences vidéo a été randomisé. En effet, si les sujets visionnent une quantité importante de séquences vidéo dégradées de score «médiocre», ensuite, dès qu'ils regardent une séquence de score «moyen», ils peuvent donner la note «bonne».

### Les sujets

Le nombre de sujets recommandés par l'UIT-T[6] est de 4 à 40. Dans la pratique, un minimum de 15 à 18 de visionneurs non-experts devrait être utilisé. Nous avons invité 29 personnes à réaliser le test subjectif. Tous les spectateurs étaient des étudiants de différentes disciplines âgés de 17 à 40 ans avec peu ou aucune expérience de ce type de tests. Chaque utilisateur a testé les 81 échantillons en utilisant la méthode ACR pour évaluer la qualité.

### Instructions aux utilisateurs

Les tests subjectifs sont composés de deux phases: la phase initiale (les instructions et la phase de formation) et des sessions de tests. Les sujets ont été sensibilisés sur l'importance du test et le sérieux demandé dans leur jugement. Une session de formation sur l'application a été incluse avant le test réel. Le but de cette session était d'expliquer pleinement ce qui est requis par les utilisateurs pour les familiariser à ce type de test. Nous avons assisté les utilisateurs durant les premières séquences de tests, ensuite nous les avons laissés continuer seul dès qu'ils commençaient à se familiariser avec le test. La structure générale est expliquée dans la figure 12.

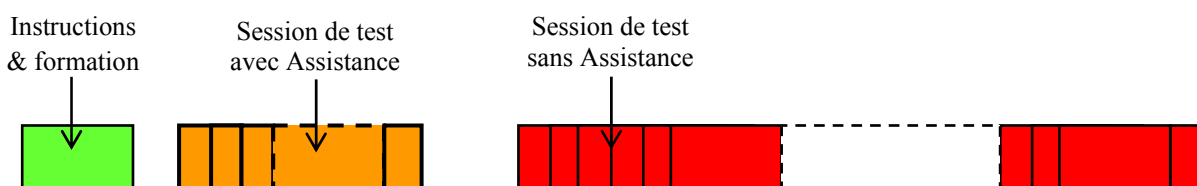


Figure- 12 Structure générale du test subjectif

### Evaluation

Nous avons employé la méthode ACR pour évaluer la qualité de 81 échantillons en utilisant l'échelle de notes à 5 points et en se conformant aux recommandations de l'UIT-R [15]. La méthode de notation a été clairement illustrée au cours de la phase de formation. En outre, après la visualisation de chaque séquence vidéo, les utilisateurs avaient juste à cliquer sur un *bouton radio* pour donner un score sur la qualité (figure 13). Ensuite, les utilisateurs devaient cliquer sur le *bouton continuer* pour afficher une nouvelle séquence. Cette opération était répétée 81 fois pour chaque utilisateur. Les vidéos sélectionnées étaient de courte durée (d'environ 10 secondes) et la phase de notation était moins de 10 secondes. La séance de test et de formation a duré à peu près de 22 à 27 minutes.

Excellente	<input type="radio"/>
Bonne	<input checked="" type="radio"/>
Assez bonne	<input type="radio"/>
Pauvre	<input type="radio"/>
Médiocre	<input type="radio"/>

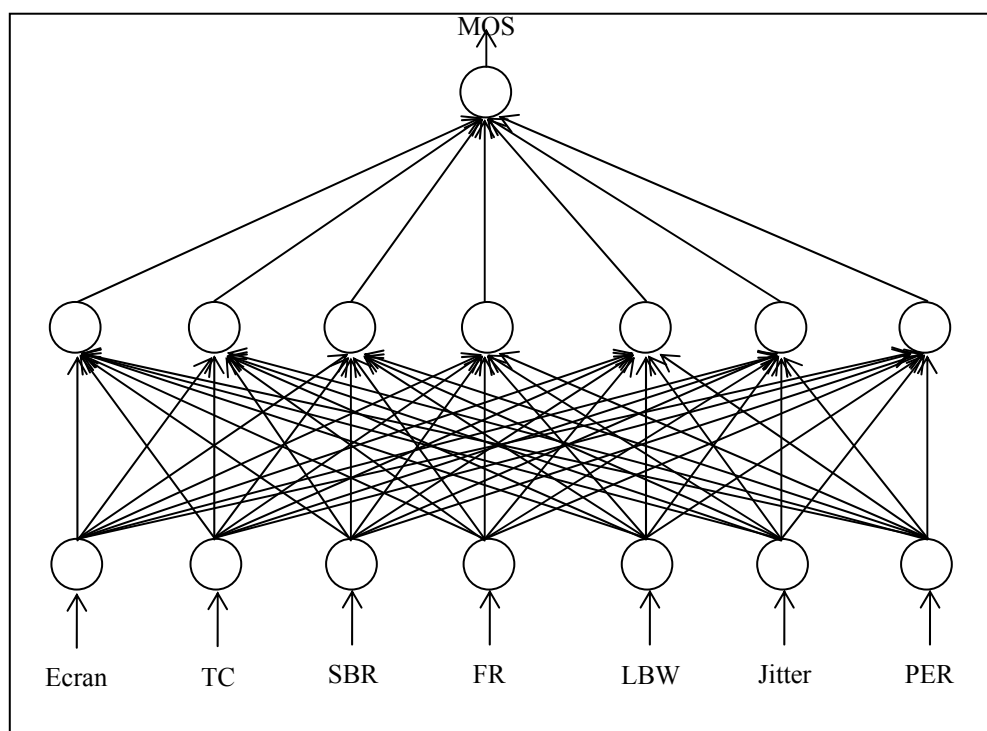
Figure-13. Échelle d'évaluation du test subjectif

### 3.4. Résultats

Après la fin des tests subjectifs qui ont duré une semaine, nous avons recueilli 2349 évaluations des 81 échantillons. Ensuite, des analyses et filtrages statistiques ont été effectués pour identifier les sujets qui ne pouvaient pas fournir des estimations fiables suivant la description de la section 3.2.2. En conséquence, nous avons rejeté toutes les notes de quatre sujets. Enfin, la valeur de MOS pour chaque échantillon a été calculée suivant les recommandations de l'UIT-T [6]. Ensuite, nous avons regroupé les 81 échantillons des trois écrans dans une seule base. La nouvelle base de données regroupe 243 échantillons. Voir les résultats dans le tableau A.2 en annexe.

#### 3.4.1 Architecture du réseau de neurones

Nous avons utilisé un réseau de neurones aléatoire (RNN) à trois couches en employant l'architecture *feedforward*. Le nombre de neurones dans la couche d'entrée est de sept correspondant au nombre de paramètres choisis. Il y a un seul neurone en sortie qui correspond à la valeur du MOS. L'algorithme d'apprentissage pour l'RNN est le célèbre algorithme « *gradient descent* ». Le nombre de neurones cachés est de 7. La figure 14 représente l'architecture de notre RNN.



**Figure-14** Architecture Feed-forward du RNN

Nous avons divisé notre BD de 243 échantillons en deux ensembles : un contenant 219 échantillons pour entraîner l'RNN, et l'autre contenant 24 autres échantillons pour la phase de validation. Après l'apprentissage de notre RNN, nous avons obtenu un facteur de corrélation de 0.9213, et une erreur quadratique moyenne de 0.004 ; cela implique que le RNN s'adapte bien à la manière dont les humains ont évalué la qualité de la vidéo. Le résultat est montré sur les figures 15 et 16.

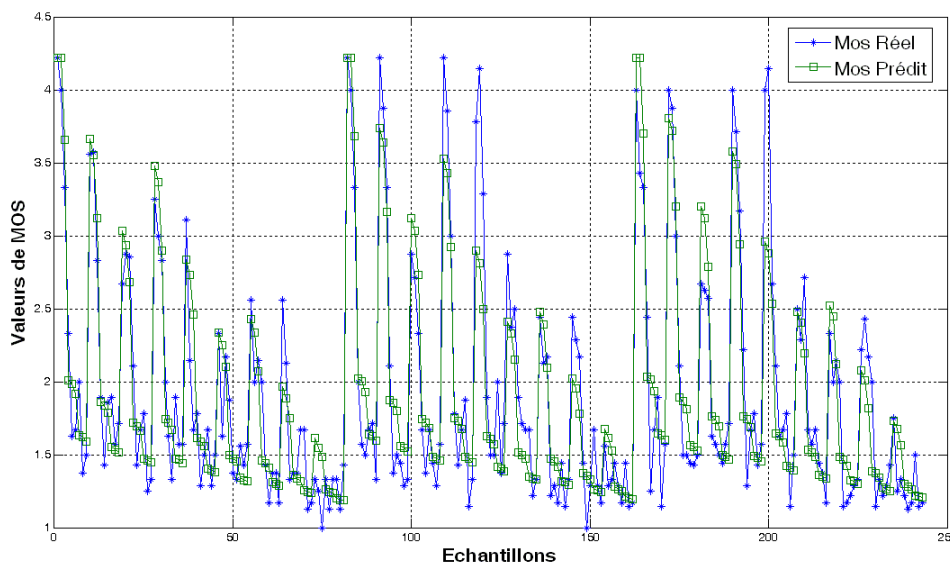


Figure-15 Corrélation entre les valeurs de MOS réelles et prédites de la DB d'apprentissage.

### 3.4.2 Comportement de réseaux de neurones

Nous avons testé l'RNN avec la BD qui contient des échantillons inconnus et qu'il ne les a jamais vus pendant le processus d'apprentissage. Nous avons obtenu des résultats avec un coefficient de corrélation de 0.8667 et une erreur quadratique moyenne de 0.004. Comme on peut le voir dans la figure 13, les performances du RNN sont bonnes. Nous illustrons également des nuages de points pour cet exemple de test dans la figure 13. On constate qu'une fois que le RNN est bien entraîné, il peut estimer les résultats même si les entrées ne sont pas nécessairement des échantillons employés pour l'entraîner. Par conséquent, les RNN peuvent estimer la qualité subjective de la vidéo de façon similaire aux sujets humains.

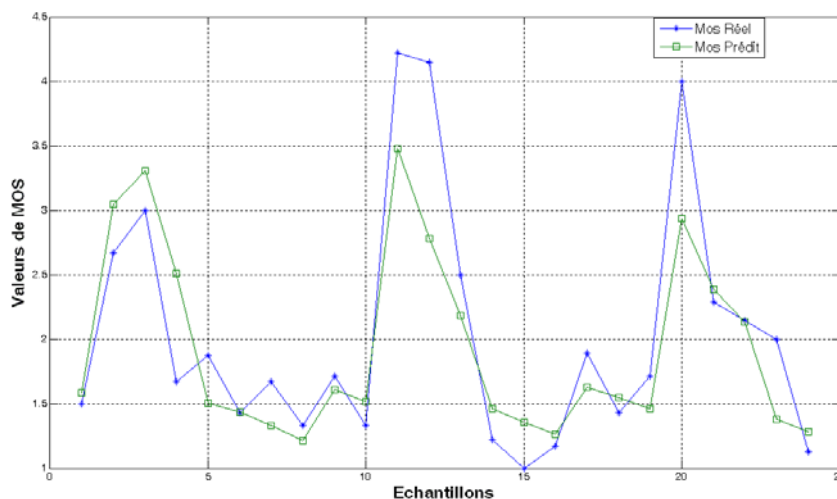


Figure-16 Corrélation entre les valeurs de MOS réelles et prédites de la DB de test.

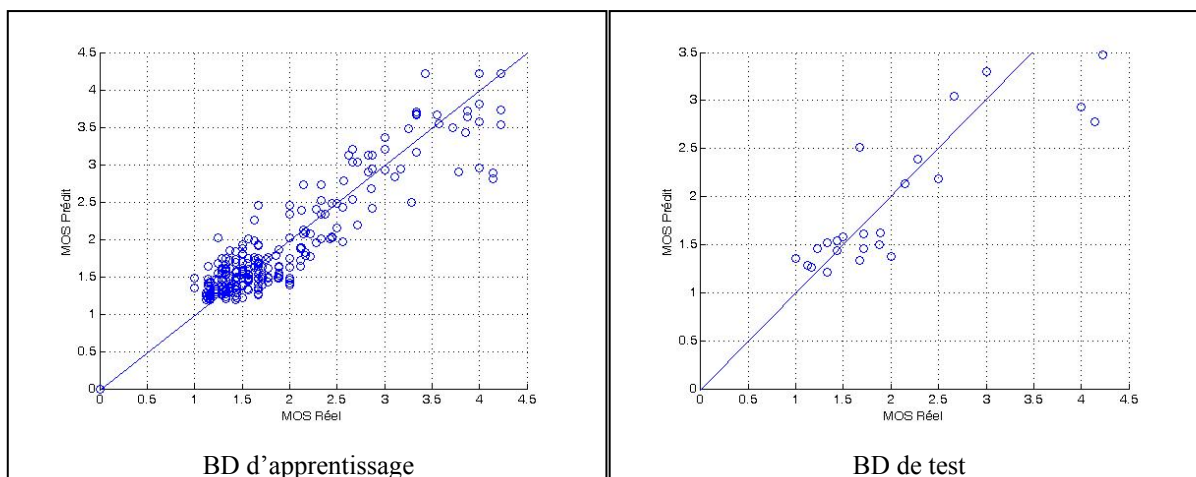


Figure-17 Corrélation entre les valeurs réelles et prédites de MOS.

### 3.4.3 Impact des paramètres sur la qualité vidéo

Dans cette section, nous étudions et analysons les effets des sept paramètres de la QoS sur la QoE. À cet effet, un RNN qualifié est employé afin d'estimer la qualité. Nous avons étudié les paramètres deux à deux, en outre nous avons fait varier les deux paramètres en gardant les cinq autres fixés sur les valeurs du tableau 5. La MOS est utilisée comme un indicateur pour mesurer la qualité.

Des représentations en trois dimensions ont été choisies pour observer l'influence des deux paramètres sélectionnés sur la QoE.

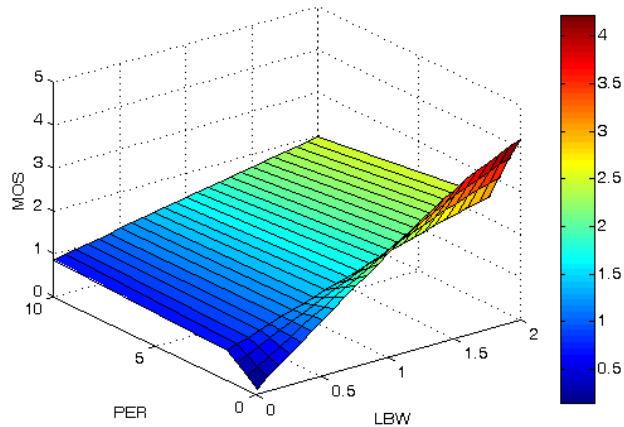
Ecran	Séquences	SBR(kbps)	FR(fps)	LWB(mbps)	PER(%)	Jitter(ms)
Ecran de PC	vidéo1(P.M)	1000 kbps	30	2 mbps	0%	0ms

Tableau 5: Valeurs fixe des paramètres non observés

#### 3.4.3.1 Impact de LBW et de PER sur la qualité

Pour évaluer l'impact de la bande passante (LBW) et le taux de perte des paquets (PER) sur la qualité vidéo (MOS), on a fait varier la bande passante de 0 à 2mbps en prenant cinq valeurs (0, 0.5, 1, 1.5 et 2mbps) et le taux de perte de 0 à 10% (0, 2.5, 5, 7.5 et 10%). Les cinq autres paramètres ont été maintenus aux valeurs fixées sur le tableau 5.

La figure 18 montre les valeurs de MOS estimées par notre RNN. On observe que la qualité diminue lorsqu'on augmente le PER de 0 jusqu'à 10%. De plus, on remarque que le PER a un impact direct sur la qualité dès que sa valeur dépasse 2.5 %. Cela dépend aussi des valeurs du LBW, avec l'augmentation de la bande passante, le PER perd un peu de son influence sur la qualité, mais reste qu'il a un très grand effet.

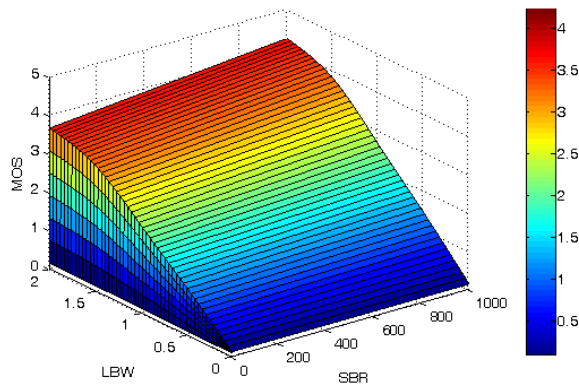


**Figure-18** Impact de PER et LBW sur la MOS.

En tenant compte des paramètres du tableau 5, la qualité vidéo est bonne pour une bande passante dépassant les 1.5mbps et un taux de perte inférieure ou égale à 2.5%, mais elle se dégrade très rapidement avec une perte de paquets dépassant ce seuil.

### 3.4.3.2 Impact de SBR et LBW sur la qualité

Afin d'évaluer l'impact du débit de flux sortant de l'encodeur (SBR) et de la bande passante (LBW) sur la qualité vidéo. Nous avons fixé les autres paramètres sur les valeurs du tableau 5 et varier les SBR de 0 à 1000 kbps (0, 250, 500, 750 et 1000kbps) et le LBW de 0 à 2mbps. La figure 19 montre les valeurs de MOS en fonction de SBR et LBW



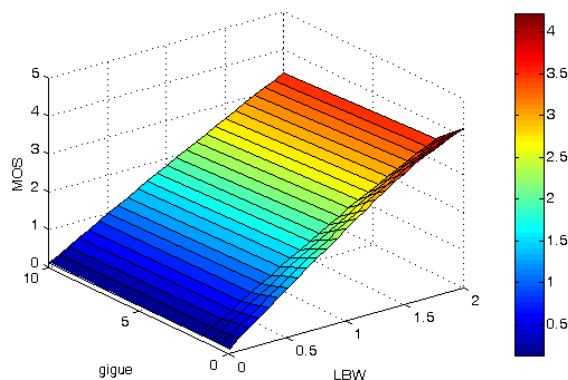
**Figure-19** Impact de SBR et LBW sur la MOS.

On observe que la bande passante (LBW) améliore la qualité si la vidéo est codée avec un débit (SBR) inférieur à LBW. Cependant, une dégradation de la qualité est observable si le débit expéditeur (SBR) est plus important que la bande passante (LBW).

### 3.4.3.3 Impact de la gigue et LBW sur la qualité

Pour évaluer l'impact de la gigue et de la bande passante (LBW) sur la qualité vidéo. Nous avons fait varier la gigue de 0 à 10ms (0, 2.5, 5, 7.5 et 10ms) et le LBW de 0 à 2mbps.

Les autres paramètres sont fixés sur les valeurs du tableau 5. La figure 20 montre les valeurs de MOS en fonction de la gigue et LBW.

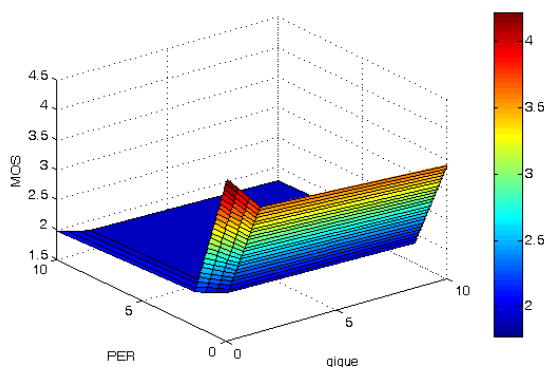


**Figure-20** Impact de la gigue et LBW sur la MOS.

L'influence de la gigue sur la qualité est semblable à celle du PER mais à un taux moins important. La gigue a un impact sur la qualité si elle dépasse les 5ms avec une LBW de 1.5mbps. Toutefois, on peut avoir une bonne qualité avec une gigue allant jusqu'à 7.5 ms si la LBW est à 2mbps.

### 3.4.3.3 Impact de la gigue et PER sur la qualité

Pour évaluer l'impact de la gigue et de taux de perte (PER) sur la qualité vidéo. Nous avons fait varier la gigue de 0 à 10ms (0, 2.5, 5, 7.5 et 10ms) et le PER de 0 à 10% (0, 2.5, 5, 7.5 et 10%). Les autres paramètres sont fixés sur les valeurs du tableau 5. La figure 21 montre les valeurs de MOS en fonction de la gigue et PER.



**Figure-21** Impact de la gigue et PER sur la MOS.

Ce test est très important pour connaître l'influence de la gigue et du taux de perte ensemble sur la QoE d'une part, et d'autre part, pour savoir quel est le paramètre qui le plus grand impact. En outre, on remarque que le PER a une influence beaucoup plus importante que la gigue. En effet, même avec une gigue qui atteint 10ms on a une qualité presque bonne. Ce n'est pas le cas pour le PER, qui dégrade la qualité (inférieur à 2) malgré une gigue qui vaut 0ms.



### 3.5 Conclusions

Nous avons réalisé un modèle pour évaluer automatiquement la QoE pour les applications de flux vidéo en temps réel en fonction de la QoS. En tenant compte du fait qu'il y a beaucoup de facteurs affectant la qualité, en particulier les déformations dues au codage et la communication sur des réseaux de paquet, les avantages de notre modèle sont les suivantes : (1) les résultats obtenus se corrèlent bien avec la perception humaine, car notre procédé est établi en utilisant des données subjectives ; (2) il est rapide en termes de temps de calcul, car un réseau de neurones aléatoire (RRN) bien entraîné est employé pour mesurer la qualité ; (3) il n'y a aucun besoin d'accéder aux signaux originaux de la vidéo. Nous avons employé les recommandations de l'UIT-T pour évaluer subjectivement la qualité vidéo en tenant compte de plusieurs paramètres vidéo, de codage, de réseau et d'affichage. Les paramètres considérés sont : le débit (SBR), le taux d'images par seconde (FR), la bande passante (LBW), le taux de perte de paquet (LR), la variation dans le délai (Jitter), le type de contenu vidéo (CT) et le type d'écran. On a généré une base de données contenant l'évaluation subjective de 25 sujets sur une combinaison des différents paramètres qu'on vient de cité. Cette base de données sera utilisée dans le futur par le laboratoire pour des études plus étendues. L'application de la méthodologie choisie a mené à des résultats qui se corrèlent bien (0,9213) avec l'évaluation des sujets humains. Le fait que la QoE puisse être estimée avec une grande confiance, permet de développer de nouveaux protocoles du réseau de transmission basé sur cette estimation.

### 3.6 Perspectives

Nous décrivons dans qui suivent certaines extensions possibles du travail présenté.

- **Évaluation du multimédia.** La qualité du multimédia regroupe la qualité vidéo et de l'audio. Nous avons remarqué lors de nos expériences que le son de la vidéo disparaissait dès que le taux de perte atteignait les 5% ou que la gigue égalait 10ms. Il serait intéressant d'étendre le travail pour évaluer le son et la vidéo ensemble en intégrant de nouveau paramètres, les variations sur l'encodage audio par exemple.
- **Extension des paramètres réseau.** Les effets combinés d'autres facteurs peuvent être étudiés. On a vu que le taux de perte de paquets dans le réseau est l'un des facteurs les plus importants affectant la qualité vidéo. Le taux de perte était appliqué uniformément sur les paquets. Par ailleurs, il existe des moyens pour contrôler la distribution de ce taux, c'est ce qu'on appelle le taux de perte conditionnel. Il est important d'étudier l'impact de la distribution de la perte sur la qualité. De plus, le profil utilisateur n'a pas été pris en compte dans notre projet, pour des tests subjectifs sur une large échelle, on peut ajouter le facteur humain et classer les sujets humains selon leurs âges, sexe, niveau visuel, taux de visualisation de vidéo par jour ... .
- **Topologies du réseau.** Les paramètres de QoS étaient variés sur un lien direct entre le serveur et les clients. Il serait intéressant de faire des tests subjectifs sur un réseau réel, mais sur lequel on a un contrôle sur ses composantes (réseau reliant les universités par exemple). Par ailleurs, le développement d'applications en temps réel et multimédia dans les réseaux sans fil se développe de plus en plus de nos jours. Il serait important d'étudier les effets de ce type de réseau sur la QoE.
- **Conception d'un classifieur de contenu.** On propose dans [1] une méthode pour classifier les contenus vidéos selon leurs caractéristiques spatio-temporelles. Un résultat important de ce travail est que les auteurs avaient utilisé le classifieur pour mesurer la complexité du mouvement des séquences vidéo. Nous croyons que

l'utilisation d'un tel outil permettra d'avoir des estimations de la QoE sur des mesures qui reflète la complexité du mouvement du contenu vidéo en temps réel.

# Chapitre 4

## Conclusion

### 4.1 Bilan technique

D'un point de vue scientifique, ce stage m'a beaucoup apporté et m'apportera encore beaucoup certainement dans les jours à venir. J'ai eu la chance de côtoyer et de discuter tous les jours avec des chercheurs pleinement passionnés par leur sujet. Je pouvais demander des conseils à n'importe qui pour bénéficier de son savoir dans son domaine de compétence, c'est un apport inestimable pour l'amélioration de mes compétences.

Du côté technique, le bilan est positif. J'ai approfondi mes connaissances et mon utilisation du SGBD MySQL et du langage PHP. J'ai aussi consolidé ma maîtrise de certains outils comme NetEM, MATLAB et j'en ai découvert d'autres comme Weka (très utile pour différencier les résultats de différents classifieurs) par exemple.

D'un point de vue méthodologique, j'ai appris ce que les réseaux de neurones et comment les manier, les comprendre et me poser des questions sur les paramètres à choisir selon la précision de l'estimateur que je souhaite obtenir. C'est une partie très importante de la réflexion et j'ai compris que cela ne sert à rien de se plonger dans le code avant d'avoir bien ciblé les problèmes et les enjeux.

### 4.2 Bilan humain

Étant un étudiant étranger, cette expérience en France a été enrichissante à plus d'un titre. Le bilan humain est également très positif. Je suis heureux d'avoir rencontré de nouvelles personnes, de cultures diverses et variées. Ce genre d'expériences contribue au développement de soi-même et à l'ouverture d'esprit. J'ai aussi remarqué que la communication était essentielle pour la réalisation d'un projet. Ainsi, bien formuler ses questions, ses remarques et ses idées est quelque chose de fondamental, en particulier en recherche.

### 4.3 Bilan du projet

La difficulté de la tâche était à la hauteur de ce que j'attendais et c'est très bénéfique de se confronter à de tels problèmes. J'ai pris conscience de la nécessité d'avoir un grand nombre de données pour l'apprentissage du réseau de neurones afin d'avoir un bon estimateur.

J'ai trouvé le stage très intéressant et il me faisait faire appel à des connaissances acquises durant l'année universitaire, ce qui est un bon moyen de les entretenir. Je me suis rendu compte qu'un chercheur doit tout d'abord connaître l'ensemble des outils qu'il a à sa disposition. Ensuite, faire appel à ces outils au moment que les conditions du problème l'imposent. Enfin, il doit élargir son domaine de compétence et ajouter sa propre réflexion aux problèmes qu'il a à résoudre.

J'ai également réalisé la difficulté et le travail qu'il faut apporter pour obtenir des résultats satisfaisants. Il faut être organisé, méthodique et toujours conscient des objectifs qu'on doit accomplir.

# Bibliographies

- [1] A. Khan. *Video quality prediction for video over wireless access networks (UMTS and WLAN)*. Doctor of Philosophy Thesis, University of Plymouth, December 2011
- [2] F. Boulos. *Transmission d'images et de vidéos sur réseaux à pertes de paquets : mécanismes de protection et optimisation de la qualité perçue*. Thèse de Doctorat, de l'Université de Nantes, février 2010
- [3] M.V. Rico. *Évaluation Pseudo-Subjective de la Qualité d'un Flux Multimédia et ses Applications au Contrôle*. Thèse de Doctorat, de l'Université de Rennes 1, Novembre 2005
- [4] D. Norman and S. Draper, *User centered system design: New perspectives on human-computer interaction*. L. Erlbaum Associates, 1986
- [5] S. R. Gulliver and G. Ghinea. The Perceptual and Attentive Impact of Delay and Jitter in Multimedia Delivery. *IEEE Transactions on Broadcasting*, 53(2) :pp. 449–458, juin 2007.
- [6] ITU-T Recommendation P.800.  
*Methods for subjective determination of transmission quality*.  
<http://www.itu.int/>.
- [7] H.R. Wu, C. Lambrecht, M. Yuen, and B. Qiu.  
Quantitative quality and impairment metrics for digitally coded images and image sequences.  
In *Proceedings of Australian Telecommunication Networks & Applications Conference*, Dec. 1996.
- [8] H.R. Wu, T. Ferguson, and B. Qiu.  
Digital video quality evaluation using quantitative quality metrics.  
In *Proceedings of 4th International Conference on Signal Processing*, 1998.
- [9] S. Dimolitsas.  
Objective speech distortion measures and their relevance to speech quality assessments.  
*IEEE Proceedings*, 136(5), Oct. 1989.
- [10] J.H. Hansen and B.L. Pellom.  
Speech enhancement and quality assessment: A survey.  
In *submitted to IEEE Sig. Proc. Mag.*, Nov. 1998.
- [11] S.R. Quackenbush, T.P. Barnwell, and M.A. Clements.  
*Objective measures of speech quality*.  
Prentice Hall, New Jersey, 1988.
- [12] Mathias Gjerstad Lervold, Liyuan Xing, Andrew Perkis. “Quality of experience in internet television”.
- [13] Kuan-Ta Chen, Chen-Chi Wu, Yu-Chun Chang and Chin-Laung Lei. “QoE Evaluation Framework for Multimedia Content”.
- [14] ITU-T Recommendation P.911.  
*Subjective audiovisual quality assessment methods for multimedia applications*.  
<http://www.itu.int/>.
- [15] ITU-R Recommendation BT.500-10.  
*Methodology for the subjective assessment of the quality of television pictures*.  
<http://www.itu.int/>.
- [16] J. Aguilar and A. Colmenares. *Recognition algorithm using evolutionary learning on the random neural network*. Proceedings of the 1997 IEEE International Conference on Neural Networks, 2 :1023\_1028, juin 1997.
- [17] J. Aguilar and V. Rossell. *Multiple classes random neural network model and color pattern recognition problems*. IEEE-INNS-ENNS International Joint Conference on Neural Networks (IJCNN'00), 5, 2000.
- [18] C. Hubert. Pattern completion with the random neural network using the rprop learning algorithm. *Systems, Man and Cybernetics*, 1993. 'Systems Engineering in the Service of Humans', *Conference Proceedings.*, 2 :613\_617, octobre 1993.
- [19] E. Gelenbe, E. Seref, and Z. Xhu. *Towards networks with intelligent packets*. Proceedings of IEEE-ICTAI Conference on Tools for Artificial Intelligence, pages 47\_54, 1999.
- [20] A. Cornuéjols and L. Miclet. *Apprentissage Artificiel*, Concepts et Algorithmes. Eyrolles, 2002.

# Annexe

*Tableau A.1 Base des combinaisons*

Combinaison	Vidéo	SBR	FR	LBW	JITTER	PER
1	1	1000	30	2mbit	0ms	0
2	1	1000	15	2mbit	0ms	5
3	1	1000	10	2mbit	0ms	10
4	1	800	30	2mbit	5ms	0
5	1	800	15	2mbit	5ms	5
6	1	800	10	2mbit	5ms	10
7	1	600	30	2mbit	10ms	0
8	1	600	15	2mbit	10ms	5
9	1	600	10	2mbit	10ms	10
10	1	1000	30	1.5mbit	0ms	0
11	1	1000	15	1.5mbit	0ms	5
12	1	1000	10	1.5mbit	0ms	10
13	1	800	30	1.5mbit	5ms	0
14	1	800	15	1.5mbit	5ms	5
15	1	800	10	1.5mbit	5ms	10
16	1	600	30	1.5mbit	10ms	0
17	1	600	15	1.5mbit	10ms	5
18	1	600	10	1.5mbit	10ms	10
19	1	1000	30	1mbit	0ms	0
20	1	1000	15	1mbit	0ms	5
21	1	1000	10	1mbit	0ms	10
22	1	800	30	1mbit	5ms	0
23	1	800	15	1mbit	5ms	5
24	1	800	10	1mbit	5ms	10
25	1	600	30	1mbit	10ms	0
26	1	600	15	1mbit	10ms	5
27	1	600	10	1mbit	10ms	10
28	2	1000	30	2mbit	0ms	0
29	2	1000	15	2mbit	0ms	5
30	2	1000	10	2mbit	0ms	10
31	2	800	30	2mbit	5ms	0
32	2	800	15	2mbit	5ms	5
33	2	800	10	2mbit	5ms	10
34	2	600	30	2mbit	10ms	0
35	2	600	15	2mbit	10ms	5
36	2	600	10	2mbit	10ms	10
37	2	1000	30	1.5mbit	0ms	0

38	2	1000	15	1.5mbit	0ms	5
39	2	1000	10	1.5mbit	0ms	10
40	2	800	30	1.5mbit	5ms	0
41	2	800	15	1.5mbit	5ms	5
42	2	800	10	1.5mbit	5ms	10
43	2	600	30	1.5mbit	10ms	0
44	2	600	15	1.5mbit	10ms	5
45	2	600	10	1.5mbit	10ms	10
46	2	1000	30	1mbit	0ms	0
47	2	1000	15	1mbit	0ms	5
48	2	1000	10	1mbit	0ms	10
49	2	800	30	1mbit	5ms	0
50	2	800	15	1mbit	5ms	5
51	2	800	10	1mbit	5ms	10
52	2	600	30	1mbit	10ms	0
53	2	600	15	1mbit	10ms	5
54	2	600	10	1mbit	10ms	10
55	3	1000	30	2mbit	0ms	0
56	3	1000	15	2mbit	0ms	5
57	3	1000	10	2mbit	0ms	10
58	3	800	30	2mbit	5ms	0
59	3	800	15	2mbit	5ms	5
60	3	800	10	2mbit	5ms	10
61	3	600	30	2mbit	10ms	0
62	3	600	15	2mbit	10ms	5
63	3	600	10	2mbit	10ms	10
64	3	1000	30	1.5mbit	0ms	0
65	3	1000	15	1.5mbit	0ms	5
66	3	1000	10	1.5mbit	0ms	10
67	3	800	30	1.5mbit	5ms	0
68	3	800	15	1.5mbit	5ms	5
69	3	800	10	1.5mbit	5ms	10
70	3	600	30	1.5mbit	10ms	0
71	3	600	15	1.5mbit	10ms	5
72	3	600	10	1.5mbit	10ms	10
73	3	1000	30	1mbit	0ms	0
74	3	1000	15	1mbit	0ms	5
75	3	1000	10	1mbit	0ms	10
76	3	800	30	1mbit	5ms	0
77	3	800	15	1mbit	5ms	5
78	3	800	10	1mbit	5ms	10
79	3	600	30	1mbit	10ms	0
80	3	600	15	1mbit	10ms	5
81	3	600	10	1mbit	10ms	10

Tableau A.2 Base des MOS

Combinaison	Ecran	Vidéo	SBR	FR	LBW	JITTER	PER	MOS(réel)	Mos(Prédit)
1	Ecran PC	1	1000	30	2mbit	0ms	0	4,22222222	4,22222222
1	Ecran Samsung	1	1000	30	2mbit	0ms	0	4	4,22222222
1	Netbook Asus	1	1000	30	2mbit	0ms	0	3,33333333	3,65897778
2	Ecran PC	1	1000	15	2mbit	0ms	5	2,33333333	2,01062222
2	Ecran Samsung	1	1000	15	2mbit	0ms	5	1,625	1,98655556
2	Netbook Asus	1	1000	15	2mbit	0ms	5	1,66666667	1,919
3	Ecran PC	1	1000	10	2mbit	0ms	10	2	1,63104444
3	Ecran Samsung	1	1000	10	2mbit	0ms	10	1,375	1,61795556
3	Netbook Asus	1	1000	10	2mbit	0ms	10	1,5	1,59177778
4	Ecran PC	1	800	30	2mbit	5ms	0	3,55555556	3,66362222
4	Ecran Samsung	1	800	30	2mbit	5ms	0	3,57142857	3,55088889
4	Netbook Asus	1	800	30	2mbit	5ms	0	2,83333333	3,12275556
5	Ecran PC	1	800	15	2mbit	5ms	5	1,88888889	1,85862222
5	Ecran Samsung	1	800	15	2mbit	5ms	5	1,42857143	1,83413333
5	Netbook Asus	1	800	15	2mbit	5ms	5	1,85714286	1,786
6	Ecran PC	1	800	10	2mbit	5ms	10	1,88888889	1,54955556
6	Ecran Samsung	1	800	10	2mbit	5ms	10	1,57142857	1,53604444
6	Netbook Asus	1	800	10	2mbit	5ms	10	1,71428571	1,51704444
7	Ecran PC	1	600	30	2mbit	10ms	0	2,66666667	3,0362
7	Ecran Samsung	1	600	30	2mbit	10ms	0	2,875	2,93655556
7	Netbook Asus	1	600	30	2mbit	10ms	0	2,85714286	2,67984444
8	Ecran PC	1	600	15	2mbit	10ms	5	2,11111111	1,7214
8	Ecran Samsung	1	600	15	2mbit	10ms	5	1,42857143	1,69691111
8	Netbook Asus	1	600	15	2mbit	10ms	5	1,66666667	1,66524444
9	Ecran PC	1	600	10	2mbit	10ms	10	1,77777778	1,47397778
9	Ecran Samsung	1	600	10	2mbit	10ms	10	1,25	1,46004444
9	Netbook Asus	1	600	10	2mbit	10ms	10	1,33333333	1,4478
10	Ecran PC	1	1000	30	1.5mbit	0ms	0	3,25	3,48122222
10	Ecran Samsung	1	1000	30	1.5mbit	0ms	0	3	3,36426667
10	Netbook Asus	1	1000	30	1.5mbit	0ms	0	2,83333333	2,89897778
11	Ecran PC	1	1000	15	1.5mbit	0ms	5	2	1,74166667
11	Ecran Samsung	1	1000	15	1.5mbit	0ms	5	1,625	1,71886667
11	Netbook Asus	1	1000	15	1.5mbit	0ms	5	1,33333333	1,67073333
12	Ecran PC	1	1000	10	1.5mbit	0ms	10	1,88888889	1,47524444
12	Ecran Samsung	1	1000	10	1.5mbit	0ms	10	1,57142857	1,463
12	Netbook Asus	1	1000	10	1.5mbit	0ms	10	1,57142857	1,44357778
13	Ecran PC	1	800	30	1.5mbit	5ms	0	3,11111111	2,83395556
13	Ecran Samsung	1	800	30	1.5mbit	5ms	0	2,14285714	2,73262222
13	Netbook Asus	1	800	30	1.5mbit	5ms	0	1,66666667	2,45944444
14	Ecran PC	1	800	15	1.5mbit	5ms	5	1,77777778	1,61331111
14	Ecran Samsung	1	800	15	1.5mbit	5ms	5	1,28571429	1,59008889
14	Netbook Asus	1	800	15	1.5mbit	5ms	5	1,5	1,55842222

15	Ecran PC	1	800	10	1.5mbit	5ms	10	1,66666667	1,40515556
15	Ecran Samsung	1	800	10	1.5mbit	5ms	10	1,28571429	1,39248889
15	Netbook Asus	1	800	10	1.5mbit	5ms	10	1,5	1,3794
16	Ecran PC	1	600	30	1.5mbit	10ms	0	2,33333333	2,337
16	Ecran Samsung	1	600	30	1.5mbit	10ms	0	1,625	2,25002222
16	Netbook Asus	1	600	30	1.5mbit	10ms	0	2,16666667	2,09886667
17	Ecran PC	1	600	15	1.5mbit	10ms	5	1,875	1,49677778
17	Ecran Samsung	1	600	15	1.5mbit	10ms	5	1,375	1,47397778
17	Netbook Asus	1	600	15	1.5mbit	10ms	5	1,33333333	1,4554
18	Ecran PC	1	600	10	1.5mbit	10ms	10	1,55555556	1,33971111
18	Ecran Samsung	1	600	10	1.5mbit	10ms	10	1,42857143	1,32662222
18	Netbook Asus	1	600	10	1.5mbit	10ms	10	1,57142857	1,31944444
19	Ecran PC	1	1000	30	1mbit	0ms	0	2,55555556	2,43115556
19	Ecran Samsung	1	1000	30	1mbit	0ms	0	2	2,33573333
19	Netbook Asus	1	1000	30	1mbit	0ms	0	2,14285714	2,07226667
20	Ecran PC	1	1000	15	1mbit	0ms	5	2	1,46046667
20	Ecran Samsung	1	1000	15	1mbit	0ms	5	1,42857143	1,4402
20	Netbook Asus	1	1000	15	1mbit	0ms	5	1,16666667	1,40937778
21	Ecran PC	1	1000	10	1mbit	0ms	10	1,375	1,31311111
21	Ecran Samsung	1	1000	10	1mbit	0ms	10	1,375	1,30128889
21	Netbook Asus	1	1000	10	1mbit	0ms	10	1,16666667	1,2882
22	Ecran PC	1	800	30	1mbit	5ms	0	2,55555556	1,96797778
22	Ecran Samsung	1	800	30	1mbit	5ms	0	2,125	1,88775556
22	Netbook Asus	1	800	30	1mbit	5ms	0	1,33333333	1,748
23	Ecran PC	1	800	15	1mbit	5ms	5	1,375	1,35828889
23	Ecran Samsung	1	800	15	1mbit	5ms	5	1,375	1,33802222
23	Netbook Asus	1	800	15	1mbit	5ms	5	1,66666667	1,31944444
24	Ecran PC	1	800	10	1mbit	5ms	10	1,66666667	1,25526667
24	Ecran Samsung	1	800	10	1mbit	5ms	10	1,125	1,24344444
24	Netbook Asus	1	800	10	1mbit	5ms	10	1,16666667	1,23584444
25	Ecran PC	1	600	30	1mbit	10ms	0	1,33333333	1,615
25	Ecran Samsung	1	600	30	1mbit	10ms	0	1,25	1,54828889
25	Netbook Asus	1	600	30	1mbit	10ms	0	1	1,48368889
26	Ecran PC	1	600	15	1mbit	10ms	5	1,33333333	1,26497778
26	Ecran Samsung	1	600	15	1mbit	10ms	5	1,125	1,24471111
26	Netbook Asus	1	600	15	1mbit	10ms	5	1,33333333	1,23668889
27	Ecran PC	1	600	10	1mbit	10ms	10	1,33333333	1,20122222
27	Ecran Samsung	1	600	10	1mbit	10ms	10	1,125	1,18897778
27	Netbook Asus	1	600	10	1mbit	10ms	10	1,42857143	1,18602222
28	Ecran PC	2	1000	30	2mbit	0ms	0	4,22222222	4,22222222
28	Ecran Samsung	2	1000	30	2mbit	0ms	0	4	4,22222222
28	Netbook Asus	2	1000	30	2mbit	0ms	0	3,33333333	3,67924444
29	Ecran PC	2	1000	15	2mbit	0ms	5	2	2,0216
29	Ecran Samsung	2	1000	15	2mbit	0ms	5	1,57142857	2,00048889
29	Netbook Asus	2	1000	15	2mbit	0ms	5	1,5	1,92617778
30	Ecran PC	2	1000	10	2mbit	0ms	10	1,66666667	1,63737778



30	Ecran Samsung	2	1000	10	2mbit	0ms	10	1,71428571	1,62597778
30	Netbook Asus	2	1000	10	2mbit	0ms	10	1,33333333	1,596
31	Ecran PC	2	800	30	2mbit	5ms	0	4,22222222	3,73624444
31	Ecran Samsung	2	800	30	2mbit	5ms	0	3,875	3,63871111
31	Netbook Asus	2	800	30	2mbit	5ms	0	3,33333333	3,16328889
32	Ecran PC	2	800	15	2mbit	5ms	5	2,11111111	1,87424444
32	Ecran Samsung	2	800	15	2mbit	5ms	5	1,375	1,85271111
32	Netbook Asus	2	800	15	2mbit	5ms	5	1,5	1,79782222
33	Ecran PC	2	800	10	2mbit	5ms	10	1,44444444	1,558
33	Ecran Samsung	2	800	10	2mbit	5ms	10	1,28571429	1,54617778
33	Netbook Asus	2	800	10	2mbit	5ms	10	1,33333333	1,5238
34	Ecran PC	2	600	30	2mbit	10ms	0	2,875	3,12064444
34	Ecran Samsung	2	600	30	2mbit	10ms	0	2,71428571	3,03113333
34	Netbook Asus	2	600	30	2mbit	10ms	0	2,33333333	2,73388889
35	Ecran PC	2	600	15	2mbit	10ms	5	1,66666667	1,74166667
35	Ecran Samsung	2	600	15	2mbit	10ms	5	1,375	1,71971111
35	Netbook Asus	2	600	15	2mbit	10ms	5	1,66666667	1,68128889
36	Ecran PC	2	600	10	2mbit	10ms	10	1,44444444	1,48537778
36	Ecran Samsung	2	600	10	2mbit	10ms	10	1,28571429	1,47271111
36	Netbook Asus	2	600	10	2mbit	10ms	10	1,57142857	1,45708889
37	Ecran PC	2	1000	30	1.5mbit	0ms	0	4,22222222	3,52851111
37	Ecran Samsung	2	1000	30	1.5mbit	0ms	0	3,85714286	3,42971111
37	Netbook Asus	2	1000	30	1.5mbit	0ms	0	3	2,92135556
38	Ecran PC	2	1000	15	1.5mbit	0ms	5	1,77777778	1,75264444
38	Ecran Samsung	2	1000	15	1.5mbit	0ms	5	1,42857143	1,73237778
38	Netbook Asus	2	1000	15	1.5mbit	0ms	5	1,66666667	1,67833333
39	Ecran PC	2	1000	10	1.5mbit	0ms	10	1,875	1,48157778
39	Ecran Samsung	2	1000	10	1.5mbit	0ms	10	1,14285714	1,4706
39	Netbook Asus	2	1000	10	1.5mbit	0ms	10	1,33333333	1,4478
40	Ecran PC	2	800	30	1.5mbit	5ms	0	3,77777778	2,89897778
40	Ecran Samsung	2	800	30	1.5mbit	5ms	0	4,14285714	2,80946667
40	Netbook Asus	2	800	30	1.5mbit	5ms	0	3,28571429	2,49828889
41	Ecran PC	2	800	15	1.5mbit	5ms	5	1,88888889	1,62851111
41	Ecran Samsung	2	800	15	1.5mbit	5ms	5	1,5	1,60824444
41	Netbook Asus	2	800	15	1.5mbit	5ms	5	1,5	1,56982222
42	Ecran PC	2	800	10	1.5mbit	5ms	10	2	1,41402222
42	Ecran Samsung	2	800	10	1.5mbit	5ms	10	1,375	1,40262222
42	Netbook Asus	2	800	10	1.5mbit	5ms	10	1,71428571	1,38615556
43	Ecran PC	2	600	30	1.5mbit	10ms	0	2,875	2,41004444
43	Ecran Samsung	2	600	30	1.5mbit	10ms	0	2,375	2,33108889
43	Netbook Asus	2	600	30	1.5mbit	10ms	0	2,5	2,14826667
44	Ecran PC	2	600	15	1.5mbit	10ms	5	1,88888889	1,5162
44	Ecran Samsung	2	600	15	1.5mbit	10ms	5	1,71428571	1,49551111
44	Netbook Asus	2	600	15	1.5mbit	10ms	5	1,66666667	1,47102222
45	Ecran PC	2	600	10	1.5mbit	10ms	10	1,66666667	1,35111111
45	Ecran Samsung	2	600	10	1.5mbit	10ms	10	1,22222222	1,33928889

45	Netbook Asus	2	600	10	1.5mbit	10ms	10	1,33333333	1,32873333
46	Ecran PC	2	1000	30	1mbit	0ms	0	2,44444444	2,47633333
46	Ecran Samsung	2	1000	30	1mbit	0ms	0	2,125	2,39273333
46	Netbook Asus	2	1000	30	1mbit	0ms	0	2,16666667	2,09717778
47	Ecran PC	2	1000	15	1mbit	0ms	5	1,22222222	1,47144444
47	Ecran Samsung	2	1000	15	1mbit	0ms	5	1,28571429	1,45371111
47	Netbook Asus	2	1000	15	1mbit	0ms	5	1,16666667	1,4174
48	Ecran PC	2	1000	10	1mbit	0ms	10	1,44444444	1,31902222
48	Ecran Samsung	2	1000	10	1mbit	0ms	10	1,14285714	1,30931111
48	Netbook Asus	2	1000	10	1mbit	0ms	10	1,33333333	1,29284444
49	Ecran PC	2	800	30	1mbit	5ms	0	2,44444444	2,02371111
49	Ecran Samsung	2	800	30	1mbit	5ms	0	2,28571429	1,95151111
49	Netbook Asus	2	800	30	1mbit	5ms	0	2,16666667	1,78388889
50	Ecran PC	2	800	15	1mbit	5ms	5	1,44444444	1,37306667
50	Ecran Samsung	2	800	15	1mbit	5ms	5	1	1,35491111
50	Netbook Asus	2	800	15	1mbit	5ms	5	1,28571429	1,33126667
51	Ecran PC	2	800	10	1mbit	5ms	10	1,66666667	1,26413333
51	Ecran Samsung	2	800	10	1mbit	5ms	10	1,28571429	1,25357778
51	Netbook Asus	2	800	10	1mbit	5ms	10	1,16666667	1,2426
52	Ecran PC	2	600	30	1mbit	10ms	0	1,55555556	1,67537778
52	Ecran Samsung	2	600	30	1mbit	10ms	0	1,28571429	1,61331111
52	Netbook Asus	2	600	30	1mbit	10ms	0	1,33333333	1,52633333
53	Ecran PC	2	600	15	1mbit	10ms	5	1,44444444	1,28313333
53	Ecran Samsung	2	600	15	1mbit	10ms	5	1,28571429	1,26497778
53	Netbook Asus	2	600	15	1mbit	10ms	5	1,16666667	1,25188889
54	Ecran PC	2	600	10	1mbit	10ms	10	1,44444444	1,2122
54	Ecran Samsung	2	600	10	1mbit	10ms	10	1,14285714	1,20122222
54	Netbook Asus	2	600	10	1mbit	10ms	10	1,16666667	1,19531111
55	Ecran PC	3	1000	30	2mbit	0ms	0	4	4,22222222
55	Ecran Samsung	3	1000	30	2mbit	0ms	0	3,42857143	4,22222222
55	Netbook Asus	3	1000	30	2mbit	0ms	0	3,33333333	3,69993333
56	Ecran PC	3	1000	15	2mbit	0ms	5	2,44444444	2,03257778
56	Ecran Samsung	3	1000	15	2mbit	0ms	5	1,25	2,014
56	Netbook Asus	3	1000	15	2mbit	0ms	5	1,66666667	1,93335556
57	Ecran PC	3	1000	10	2mbit	0ms	10	1,88888889	1,64328889
57	Ecran Samsung	3	1000	10	2mbit	0ms	10	1,14285714	1,63315556
57	Netbook Asus	3	1000	10	2mbit	0ms	10	1,57142857	1,60064444
58	Ecran PC	3	800	30	2mbit	5ms	0	4	3,80422222
58	Ecran Samsung	3	800	30	2mbit	5ms	0	3,875	3,72062222
58	Netbook Asus	3	800	30	2mbit	5ms	0	3	3,20128889
59	Ecran PC	3	800	15	2mbit	5ms	5	2,11111111	1,88902222
59	Ecran Samsung	3	800	15	2mbit	5ms	5	1,5	1,87002222
59	Netbook Asus	3	800	15	2mbit	5ms	5	1,5	1,8088
60	Ecran PC	3	800	10	2mbit	5ms	10	1,44444444	1,56644444
60	Ecran Samsung	3	800	10	2mbit	5ms	10	1,42857143	1,55546667
60	Netbook Asus	3	800	10	2mbit	5ms	10	1,5	1,53013333

61	Ecran PC	3	600	30	2mbit	10ms	0	2,66666667	3,19917778
61	Ecran Samsung	3	600	30	2mbit	10ms	0	2,625	3,11937778
61	Netbook Asus	3	600	30	2mbit	10ms	0	2,57142857	2,78413333
62	Ecran PC	3	600	15	2mbit	10ms	5	1,625	1,75982222
62	Ecran Samsung	3	600	15	2mbit	10ms	5	1,57142857	1,7404
62	Netbook Asus	3	600	15	2mbit	10ms	5	1,5	1,69564444
63	Ecran PC	3	600	10	2mbit	10ms	10	1,44444444	1,49551111
63	Ecran Samsung	3	600	10	2mbit	10ms	10	1,57142857	1,48453333
63	Netbook Asus	3	600	10	2mbit	10ms	10	1,71428571	1,46553333
64	Ecran PC	3	1000	30	1.5mbit	0ms	0	4	3,57453333
64	Ecran Samsung	3	1000	30	1.5mbit	0ms	0	3,71428571	3,49135556
64	Netbook Asus	3	1000	30	1.5mbit	0ms	0	3,16666667	2,94373333
65	Ecran PC	3	1000	15	1.5mbit	0ms	5	2,22222222	1,7632
65	Ecran Samsung	3	1000	15	1.5mbit	0ms	5	1,28571429	1,74588889
65	Netbook Asus	3	1000	15	1.5mbit	0ms	5	1,66666667	1,68593333
66	Ecran PC	3	1000	10	1.5mbit	0ms	10	1,77777778	1,48791111
66	Ecran Samsung	3	1000	10	1.5mbit	0ms	10	1,42857143	1,47777778
66	Netbook Asus	3	1000	10	1.5mbit	0ms	10	1,57142857	1,45244444
67	Ecran PC	3	800	30	1.5mbit	5ms	0	4	2,9602
67	Ecran Samsung	3	800	30	1.5mbit	5ms	0	4,14285714	2,88166667
67	Netbook Asus	3	800	30	1.5mbit	5ms	0	2,66666667	2,5346
68	Ecran PC	3	800	15	1.5mbit	5ms	5	2,11111111	1,64286667
68	Ecran Samsung	3	800	15	1.5mbit	5ms	5	1,625	1,62471111
68	Netbook Asus	3	800	15	1.5mbit	5ms	5	1,66666667	1,5808
69	Ecran PC	3	800	10	1.5mbit	5ms	10	1,77777778	1,42204444
69	Ecran Samsung	3	800	10	1.5mbit	5ms	10	1,14285714	1,41191111
69	Netbook Asus	3	800	10	1.5mbit	5ms	10	1,5	1,39248889
70	Ecran PC	3	600	30	1.5mbit	10ms	0	2,5	2,47886667
70	Ecran Samsung	3	600	30	1.5mbit	10ms	0	2,28571429	2,40666667
70	Netbook Asus	3	600	30	1.5mbit	10ms	0	2,71428571	2,19428889
71	Ecran PC	3	600	15	1.5mbit	10ms	5	1,66666667	1,53393333
71	Ecran Samsung	3	600	15	1.5mbit	10ms	5	1,57142857	1,51535556
71	Netbook Asus	3	600	15	1.5mbit	10ms	5	1,66666667	1,48537778
72	Ecran PC	3	600	10	1.5mbit	10ms	10	1,44444444	1,36124444
72	Ecran Samsung	3	600	10	1.5mbit	10ms	10	1,375	1,35068889
72	Netbook Asus	3	600	10	1.5mbit	10ms	10	1,16666667	1,33717778
73	Ecran PC	3	1000	30	1mbit	0ms	0	2,33333333	2,51982222
73	Ecran Samsung	3	1000	30	1mbit	0ms	0	2	2,44762222
73	Netbook Asus	3	1000	30	1mbit	0ms	0	2,14285714	2,12166667
74	Ecran PC	3	1000	15	1mbit	0ms	5	2	1,48242222
74	Ecran Samsung	3	1000	15	1mbit	0ms	5	1,14285714	1,46637778
74	Netbook Asus	3	1000	15	1mbit	0ms	5	1,16666667	1,425
75	Ecran PC	3	1000	10	1mbit	0ms	10	1,22222222	1,32535556
75	Ecran Samsung	3	1000	10	1mbit	0ms	10	1,28571429	1,31648889
75	Netbook Asus	3	1000	10	1mbit	0ms	10	1,33333333	1,29748889
76	Ecran PC	3	800	30	1mbit	5ms	0	2,22222222	2,07691111

76	Ecran Samsung	3	800	30	1mbit	5ms	0	2,42857143	2,01188889
76	Netbook Asus	3	800	30	1mbit	5ms	0	2,16666667	1,81808889
77	Ecran PC	3	800	15	1mbit	5ms	5	2	1,387
77	Ecran Samsung	3	800	15	1mbit	5ms	5	1,14285714	1,37053333
77	Netbook Asus	3	800	15	1mbit	5ms	5	1,33333333	1,34182222
78	Ecran PC	3	800	10	1mbit	5ms	10	1,22222222	1,27215556
78	Ecran Samsung	3	800	10	1mbit	5ms	10	1,28571429	1,26244444
78	Netbook Asus	3	800	10	1mbit	5ms	10	1,42857143	1,24893333
79	Ecran PC	3	600	30	1mbit	10ms	0	1,75	1,73237778
79	Ecran Samsung	3	600	30	1mbit	10ms	0	1,25	1,67453333
79	Netbook Asus	3	600	30	1mbit	10ms	0	1,33333333	1,56686667
80	Ecran PC	3	600	15	1mbit	10ms	5	1,22222222	1,30002222
80	Ecran Samsung	3	600	15	1mbit	10ms	5	1,125	1,28355556
80	Netbook Asus	3	600	15	1mbit	10ms	5	1,16666667	1,2654
81	Ecran PC	3	600	10	1mbit	10ms	10	1,5	1,22191111
81	Ecran Samsung	3	600	10	1mbit	10ms	10	1,14285714	1,2122
81	Netbook Asus	3	600	10	1mbit	10ms	10	1,16666667	1,20375556