



HAL
open science

Mise en oeuvre d'un modèle adaptatif pour l'estimation de la qualité réelle perçue par l'utilisateur (QoE) : application aux services mobiles

Lamine Amour

► To cite this version:

Lamine Amour. Mise en oeuvre d'un modèle adaptatif pour l'estimation de la qualité réelle perçue par l'utilisateur (QoE) : application aux services mobiles. Intelligence artificielle [cs.AI]. Université Paris-Est, 2018. Français. NNT : 2018PESC0043 . tel-03122928

HAL Id: tel-03122928

<https://theses.hal.science/tel-03122928>

Submitted on 27 Jan 2021

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.



UNIVERSITÉ PARIS EST
ÉCOLE DOCTORALE MSTIC

Thèse de doctorat

Spécialité : Informatique et Génie informatique

Présentée par
Lamine AMOUR

Titre :

Mise en œuvre d'un modèle adaptatif pour
l'estimation de la qualité réelle perçue par l'utilisateur
(QoE) : Application aux services mobiles

Date de soutenance : 03/12/2018

Lieu : Département Réseaux et Télécoms & Laboratoire LiSSi, Vitry sur Seine.

Rapporteurs :

Stephane MAAG - Professeur - Télécom SudParis.
Adlen KSENTINI - Maître de Conférences des Universités HDR - Eurecom.

Directeur :

Abdelhamid MELLOUK - Professeur des Universités - UPEC.

Co-encadrant :

Sami SOUIHI - Maître de Conférences des Universités - UPEC.

Examineurs :

Thierry DIVOUX - Professeur des Universités - Université de Lorraine.
Ana Rosa CAVALLI - Professeur - Télécom SudParis.
Samia BOUZEFRANE - Maître de Conférences des Universités HDR - CNAM.

Remerciements

En plus du travail personnel, l'établissement d'une thèse dépend en grande partie des encouragements et des directives de beaucoup d'autres personnes. Cette page est peut-être la plus difficile à écrire car ceux qui ont eu une influence sur ce travail sont nombreux.

Tout d'abord, je tiens à remercier mon directeur de thèse, le Professeur Abdelhamid Mellouk, dont l'expérience et la pédagogie ont été pour moi une aide précieuse mais aussi une source de motivation et de curiosité intellectuelle durant ces dernières années. Il m'a grandement inspiré à travailler dans cette thèse et dans ma volonté à poursuivre dans le domaine de la recherche.

Je tiens également à exprimer ma sincère gratitude à mon co-encadrant, Monsieur Sami Souihi, pour les conseils et les encouragements qu'il m'a prodigués durant cette thèse. Sans ses encouragements et ses conseils, cette thèse n'aurait pas vu le jour. Particulièrement, dans les moments où le doute m'habite, il a toujours été là pour me redonner confiance et me remotiver pour aller de l'avant.

J'exprime ma profonde reconnaissance à Monsieur Stephane Maag et à Monsieur Adlen Ksentini pour l'intérêt qu'ils ont bien voulu porter à ce travail en acceptant de l'examiner et d'en être rapporteurs. Je remercie également Madame Ana Rosa Cavalli, Madame Samia Bouzefrane et Monsieur Thierry Divoux d'avoir accepté de juger mon travail.

Enfin, je tiens à souligner que j'ai eu la chance de travailler, au courant de ma thèse, avec des personnes formidables et très enrichissantes sur le plan scientifique et professionnel, ainsi que sur le plan personnel. Je tiens alors à remercier toutes ces personnes qui ont contribué de près ou de loin à l'aboutissement de cette thèse, entre autre l'ensemble des membres de laboratoire LiSSi, l'équipe Contrôle Intelligent des Réseaux (TINCNET) et les collègues de l'IUT Créteil/Vitry, qui resteront anonymes dans cette page, mais qui m'ont permis de mener à terme ce travail dans une ambiance très amicale.

Dédicaces

Je dédie cette thèse

à la mémoire de mon oncle et de ma grande mère,

à mes parents, à qui je dois beaucoup,

à ma douce femme, qui m'a supporté et beaucoup aidé,

à tous les miens, en particulier Madame et Monsieur HADDAR ainsi que le petit Dédine.

Table des matières

Remerciements	1
Dédicaces	2
Table de matières	i
Liste des Figures	iv
Liste des Tableaux	viii
Glossaire	viii
Liste des publications	xi
Résumé	xiii
Abstract	xiv
1 État de l’art de l’Expérience utilisateur (QoE)	8
1.1 Introduction	8
1.2 Motivations	9
1.3 La Qualité d’Expérience (QoE)	11
1.4 Les facteurs d’influence de la QoE	13
1.5 Évaluation de la QoE	15
1.5.1 Recommandations	16
1.5.2 Évaluation subjective	17
1.6 Modèles d’estimation de la QoE	22
1.6.1 Modèles objectifs	22

1.6.2	Métriques de l'évaluation de la qualité vidéo	24
1.6.3	Modèles hybrides	27
1.6.4	Synthèse	35
1.7	Conclusion	46
2	Campagnes subjectives pour l'évaluation de la Qualité d'Expérience usager	47
2.1	Introduction	47
2.2	Environnements de l'évaluation subjective de la QoE	48
2.2.1	Environnement contrôlé	49
2.2.2	Environnement non-contrôlé	50
2.2.3	Base de données existantes	51
2.3	Campagne d'évaluation de la QoE dans l'environnement contrôlé	53
2.3.1	Mise en œuvre de la plateforme de test	53
2.3.2	Collecte de la base de données	55
2.4	Campagne d'évaluation de la QoE dans l'environnement non-contrôlé	56
2.4.1	Projet PoQEMoN	57
2.4.2	Application de test Android	57
2.4.3	Campagne de test crowdsourcing	59
2.5	Discussion	62
2.6	Conclusion	64
3	Méthodes d'apprentissage automatique pour l'estimation de la Qualité d'Expé- rience usager	65
3.1	Introduction	65
3.2	Sélection des meilleurs facteurs impactant la Qualité d'Expérience usager	66
3.2.1	Sélection des caractéristiques	66
3.2.2	Processus de sélection	66
3.2.3	Sélection des facteurs de la Qualité d'Expérience	70
3.3	Méthodes d'apprentissage automatique	71
3.3.1	Méthodes classiques	71
3.3.2	Méthodes ensemblistes	74
3.3.3	Discussion	76
3.4	Estimation de la Qualité d'Expérience avec l'apprentissage automatique	77
3.5	Approche de sélection de facteurs impactant la QoE usager	79

3.5.1	Présentation	79
3.5.2	Formulation mathématique	80
3.6	Évaluation de l'approche de sélection de facteurs	83
3.6.1	Description	83
3.6.2	Processus	84
3.6.3	Validation	88
3.6.4	Discussion	89
3.7	Conclusion	91
4	Méthode d'estimation de la perception usager basée sur les fonctions de croyances	92
4.1	Introduction	92
4.2	Les fonctions de croyance	93
4.3	Modèle des Croyances Transférables	94
4.4	Modèle proposé : Combined Belief Functions	98
4.5	Implémentation	101
4.5.1	Étapes	101
4.5.2	Définition de la fonction de masse	103
4.5.3	Mise en œuvre de la règle pondérée de Murphy	106
4.6	Évaluation	107
4.6.1	Discussion	110
4.7	Conclusion	112
5	Méthode pour la diffusion vidéo à débit variable	113
5.1	Introduction	113
5.2	Diffusion vidéo à débit variable	114
5.2.1	Stratégies d'adaptation de la diffusion vidéo	114
5.2.2	Diffusion vidéo en flux adaptatif dynamique sur HTTP	115
5.3	Campagne de test subjective	122
5.3.1	Construction de la base de données	122
5.3.2	Sélection des facteurs	125
5.3.3	Construction du modèle	127
5.4	Solution proposée de diffusion vidéo à débit variable	128
5.4.1	Surveillance de changement de la QoE	129
5.4.2	Modèle formel	132

5.4.3	Fonctionnement	134
5.5	Évaluation	136
5.5.1	Solutions utilisées	136
5.5.2	Évaluation	140
5.5.3	Discussion	147
5.6	Conclusion	148
6	Conclusion générale et perspectives	149
6.1	Travail réalisé et contributions	149
6.2	Enseignement et retours d'expérience	151
6.3	Perspectives et travaux à venir	152

Table des figures

1	Gestion réseau orienté utilisateur	2
1.1	Facteurs impactant la QoE	13
1.2	Classification des facteurs impactant la QoE en 5 catégories	14
1.3	Méthode d'évaluation DSIS	18
1.4	Méthode d'évaluation SSCQE	19
1.5	Méthode d'évaluation DSCQS	20
1.6	Synthèse des méthodes subjectives d'évaluation de la QoE	22
1.7	Modèles objectifs selon les informations utilisées	23
1.8	Classification des méthodes objectives	24
1.9	Méthodes hybrides d'estimation de la QoE	28
1.10	Résultats des comparaisons entre les modèles puissance et logarithmique	32
2.1	Méthodes d'évaluation de la QoE	48
2.2	Campagne de test réalisée dans un environnement contrôlé	54
2.3	Captures d'écran des vidéos utilisées	56
2.4	Questions utilisées lors de la campagne de crowdsourcing	58
2.5	Campagne de test crowdsourcing	59
2.6	Emplacements testés lors de la campagne de crowdsourcing	60
2.7	Captures d'écran des vidéos utilisées lors de la campagne crowdsourcing	61
3.1	Processus de sélection des caractéristiques	67
3.2	Neurone formel	73
3.3	Fonctions d'activation	73
3.4	Comparaison des méthodes d'apprentissage automatique	77
3.5	Validation croisée	79

3.6	Approche proposée pour sélectionner les facteurs pertinents	79
3.7	Diagramme des étapes de la sélection de facteurs	82
3.8	Matrice de corrélation de 'r' Pearson	85
3.9	Exécution du processus de sélection des facteurs	88
3.10	Comparaison des approches de la sélection de facteurs	89
3.11	Distribution du taux des MOS prédits le plus proche du MOS réel	90
4.1	Mécanismes du modèle des croyances transférables	95
4.2	Étapes du modèle des croyances transférables	98
4.3	Description des étapes du modèle CBF	99
4.4	Fonctionnement du modèle CBF	101
4.5	Distribution des masses du modèle CBF	104
4.6	Performances en utilisant une taille variable de la base de données	108
4.7	Performances en utilisant la base de données complète	109
5.1	Exemple d'un déploiement typique d'un système Dash	116
5.2	Campagne de test réalisée pour la mise en application du modèle CBF	123
5.3	Matrice de corrélation utilisée pour l'application du modèle CBF	125
5.4	Architecture générale	129
5.5	Modélisation des points de changement	131
5.6	Système apprentissage par renforcement	132
5.7	Composantes de la solution	134
5.8	Diagramme de fonctionnement de la solution	136
5.9	Variations de la bande passante des traces réseaux collectées.	140
5.10	Performances en termes de nombre d'épuisement de la mémoire tampon	142
5.11	Performances en termes de changement de qualité	143
5.12	Performances en termes du total des données téléchargé	144
5.13	Performances en termes en termes de la moyenne des scores MOS prédite	145
5.14	Performances en termes des variations des scores MOS calculés	146
6.1	Détails des vidéos utilisées.	156
6.2	Distribution des scores MOS récoltés	158
6.3	Description de l'environnement contrôlé de l'évaluation de la QoE	160

Liste des tableaux

1.1	Quelques recommandations proposées par ITU-R	16
1.2	Quelques recommandations proposées par ITU-T	17
1.3	Modèles mathématiques représentant la relation entre la QoE et les facteurs	28
1.4	Valeurs des coefficients du modèle fractionnaire proposé par Wang et al.	31
1.5	Synthèse de méthodes d'estimation de la QoE utilisant la régression statistique	36
1.6	Synthèse d'études d'évaluation de la QoE usager en utilisant une approche hybride	45
2.1	Bases de données subjective de la QoE	52
2.2	Conditions générales des tests	62
3.1	Mesures utilisées par les méthodes de filtrage	69
3.2	Configurations utilisées par les méthode classiques de l'apprentissage automatique	74
3.3	Configurations utilisées par les méthode ensemblistes	76
3.4	Description des paramètres vidéo utilisés	84
3.5	Tableau comparatif de l'impact des facteurs.	88
4.1	Notations mathématiques	98
4.2	Taux des croyances résultant pour une valeur calculée du score MOS	106
4.3	Tableau comparatif des performances des méthodes utilisées	111
5.1	Comparaison des stratégies d'adaptations de service vidéo	115
5.2	Méthodes diffusion vidéo à débit variable en utilisant le standard Dash	121
5.3	Représentations utilisées dans le fichier <i>MPD</i>	124
5.4	Facteurs collectés pour mettre en application de notre proposition CBF	125
5.5	Performances de l'approche de sélection des facteurs	126
5.6	Tableau comparatif des méthodes d'estimation de la QoE	127

5.7	Comparaison des méthodes utilisées pour l'évaluation	139
5.8	Métrique utilisées pour la comparaison	142

Glossaire

ABR : Adaptive video BitRate.

ACR : Absolute Category Rating.

BF : Belief Function.

BFC : Belief Function Combination.

CDN : Content Delivery Network.

CsP : Fournisseur de service (Content service Provider).

DASH : Dynamic Adaptive Streaming over HTTP.

DMOS : Derivation of Mean Opinion Score.

ETSI : European Telecommunications Standards Institute.

FSI : Fournisseur de Service Internet.

IoT : Internet of Things.

IP : Internet Protocol.

LTE : Long Term Evolution.

MAE : Mean Absolute Error.

MCT : Modèle de Croyance Transférable.

ML : Machine Learning.

MOS : Mean Opinion Score.

MSE : Mean Squared Error.

NGN : Next Generation Networks.

NsP : Opérateur de réseau (Network service Provider).

PSNR : Peak Signal to Noise Ratio.

QUALINET : European Network on Quality of Experience in Multimedia Systems and Services.

QoE : Quality of Experience.

QoE IF : Quality of Experience Influence Factor.

QoS : Quality of Service.
RF : Random Forests.
RL : Reinforcement Learning.
SDN : Software-Defined Networking.
SLA : Service Level Agreements.
SVM : Support Vector Machine.
TIC : Technologies de l'Information et de la Communication.
UIT : Union Internationale des Télécommunications
VMAF : Video Multimethod Assessment Fusion.
VoD : Vidéo à la demande.
VoIP : Voix sur IP.
VQEG : Video Quality Experts Group.
VQM : Video Quality Metric.
WLAN : Wireless Local Area Network.

Liste des publications

Revue internationale :

1. Y. Ben Youssef, M. Afif, A. Mellouk, L. Amour and S. Tabbane. *An Ensemble Machines for QoE Prediction Model based on AHP-PCA Selection Procedure*. In *IEEE Transactions on The Journal of Mobile Communication, Computation and Information*, . pages 1 – 16. (Soumis)

Conférences et Colloques Internationaux :

1. L. Amour, S. Souihi, S. Hoceini, and A. Mellouk. *A Hierarchical Classification Model of QoE Influence Factors (QoE IFs)*. In *Wired/Wireless Internet Communications - 13th International Conference, WWIC 2015*, Malaga, Spain, pages 225 – 238, May 25 – 27, 2015.

2. L. Amour, S. Sami, S. Hoceini, and A. Mellouk. *An Open Source Platform for Perceived Video Quality Evaluation*. In *Proceedings of the 11th ACM Symposium on QoS and Security for Wireless and Mobile Networks, Q2SWinet'15*, Cancun, Mexico, pages 139 – 140, November 2015.

3. L. Amour, S. Sami, S. Hoceini, and A. Mellouk. *Building a Large Dataset for Model-based QoE Prediction in the Mobile Environment*, *Proceedings of the 18th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, MSWIM'15*, Cancun, Mexico, pages 313 – 317, November 2015.

4. L. Amour, S. Souihi, M. S. Mushtaq, S. Hoceini, and A. Mellouk. *Perceived Video Quality Evaluation based on Interactive/Repulsive relation between the QoE IFs*. In *IEEE International Conference on Communications (ICC'16)*, Kuala Lumpur, Malaysia, 23 – 27 May, 2016.

5. L. Amour, S. Souihi, M. S. Mushtaq and A. Mellouk. *QoE-based Framework to Optimize User Perceived Video Quality*, In *42nd IEEE Conference on Local Computer Networks (LCN)*,

Singapore, October 9 – 12, 2017.

6. L. Amour, M. S. Mushtaq, S. Souihi and A. Mellouk. *QoE-based Framework : Optimized Quality of DASH (OQD)*. In *Crowdsourcing and IoT Summer School (IoT Crowd 2017)*, July-August, 2017, Würzburg, Germany.

7. L. Amour, S. Souihi, M. I. Boulabiar and A. Mellouk. *An Improved QoE Estimation Method based on QoS and Affective Computing*, The 13th *IEEE International Symposium on Programming and Systems (ISPS'2018)*, Algeries, Algeria. April 24 – 26, 2018.

Conférences et Colloques Nationaux :

1. L. Amour, S. Souihi and A. Mellouk. *Proposition d'une Approche Algorithmique de Streaming adaptatif basée sur la Perception Utilisateur (QoE)*. Dans *l'école d'été GDR-RESCOM 2016 : 5G & Internet of Things (IoT)*. May 2016, Guidel plages, France.

2. L. Amour, S. Souihi and A. Mellouk, *Présentation d'un Framework QoE-DASH (QOD)*. Dans le *Colloque Automatique et Réseaux de Communication 2017 (ARC'17) du GDR CNRS MACS*, 14 Mai, 2017, Paris, France.

3. L. Amour, S. Souihi, and A. Mellouk. *Pour une Meilleure Estimation de la Qualité Perçue par l'Usager*. Dans le *Congrès National de la Recherche des IUT "CNRIUT'2018"*, Aix en Provence, France, Juin 7 – 8, 2018.

Résumé

Titre : Mise en œuvre d'un modèle adaptatif pour l'estimation de la qualité perçue par l'utilisateur (QoE) : Application aux services mobiles.

Résumé : Le contexte général de la thèse se situe dans le développement de mécanismes adaptatifs capables de récupérer des éléments d'information issus d'un environnement et d'un contexte donnés (Qualité de l'Expérience : QoE) et de s'y adapter afin d'enclencher des actions spécifiques correctrices suite à l'arrivée d'événements imprévisibles ou non souhaités comme une QoS insatisfaisante, un retour d'expérience négatif ou encore des dysfonctionnements des éléments du réseau. Pour ce faire, l'idée maîtresse défendue ici consiste à faire évoluer la gestion du réseau en migrant d'une vue "centrée réseau" où l'on se contentait uniquement des paramètres issus du réseau lui-même, vers une vue "centrée utilisateur". Cette vue consiste à considérer l'ensemble des facteurs, endogènes ou exogènes, pouvant impacter le retour d'expérience de l'utilisateur final. Les travaux développés dans le cadre de cette thèse se sont focalisés sur l'étude de ces facteurs sur l'ensemble des composants de la chaîne de traitement et de transport (utilisateur, périphérique, application, éléments du réseau) et mesurer leur impact dans l'estimation de la QoE usager. Le cas d'usage considéré concerne les approches de diffusion de vidéos en ligne. Les travaux de thèse ont conduit à la proposition d'une nouvelle solution de diffusion de vidéos à débit variable fonctionnant en boucle fermée et sensible à la QoE. Une telle solution permet d'optimiser en temps réel la qualité de la vidéo en fonction de la mesure instantanée de la QoE estimée.

Abstract

Title : QoE estimation model to Optimize User Perceived Video Quality : Mobile Services Application.

Abstract :

The general context of the thesis is the development of adaptive network mechanisms that allow to guarantee a good perceived quality of experience (QoE). The main idea consists in changing the management of the network by migrating from a "network-centric" view, where we consider only parameters from the network itself, to a "user-centric view". This view consists in considering all the factors that can impact the end-user's experience. In this thesis QoE factors are studied and their impact on the final user QoE is measured. Firstly, the subjective and objective methods used to evaluate end-user QoE are analyzed. Secondary, two subjective tests campaign, one in controlled environment (laboratory) and one in uncontrolled environment (crowd-sourcing), were performed to build subjective QoE datasets. Thirdly, a new model, called *Belief Functions Combined (BFC)*, is introduced to estimate the user QoE. This model combines the use of belief function theory, information fusion, and several machine learning (ML) methods. Within the scope of the considered use case, a new solution for adaptive bit rate video streaming (ABR) is implemented. It consists on a new closed-loop, QoE-sensitive ABR solution. Such a solution makes it possible to optimize in real time the quality of the video according to the instantaneous measurement of the estimated QoE.

Introduction générale

Contexte général de la thèse

Internet est un réseau de communication conçu initialement à des fins d'échanges simples. Son objectif premier était d'offrir aux usagers la possibilité de communiquer les uns avec les autres grâce à des services tels que la messagerie électronique et le transfert de fichiers. Néanmoins, de nos jours, une multitude de nouveaux services, comme les jeux en ligne, la vidéo à la demande (VoD) ou la téléphonie sur IP (VoIP), a vu le jour. Cette diversification, ainsi que l'augmentation de trafic qui en découle ont rendu le réseau complexe à gérer. En effet, d'après un rapport CISCO datant de 2017 [50], le trafic vidéo a connu un accroissement de 9500% sur les 15 dernières années. Ce trafic est directement lié à l'accroissement du nombre d'usagers. En outre, d'après le conseil supérieur de l'audiovisuel (CSA) en France, le marché français a atteint une progression de 91% sur la seule année 2017, avec plus de 30 millions d'usagers journaliers de la plateforme de partage de vidéo Youtube¹. Toutefois, l'ossification du réseau [169], principalement causée par l'absence d'évolution majeure du cœur de réseau et par la rigidité des équipements déployés, rend les mécanismes traditionnels de contrôle du réseau inadaptés. À cela s'ajoute la concurrence entre les fournisseurs de services qui a atteint ces dernières années un niveau sans précédent, en particulier en ce qui concerne les offres de services vidéo. Par exemple, en 2017, 65 services de la vidéo à la demande (VoD) étaient actifs en France, soit trois fois plus qu'en 2010 [41]. Ces défis et problématiques ont eu comme conséquences d'accroître les exigences et les besoins des usagers, qui souhaiteraient payer ces services à des prix plus bas

1. <https://www.omnicoreagency.com/youtube-statistics/>

en bénéficiant d'une qualité améliorée.

Dans ce contexte et afin de garantir les termes des contrats de niveaux de service (*Service Level Agreements (SLA)*), il n'est plus envisageable de concevoir un service sans intégrer les deux volets "satisfaction de l'utilisateur final" et "optimisation des ressources". Il est donc d'une importance cruciale d'estimer, ce qui est appelée communément, l'expérience usager (*Quality of Experience (QoE)*). La QoE représente la qualité réelle perçue par l'utilisateur final. L'objectif est de faire évoluer la gestion du réseau en migrant d'une vue "centrée réseau" où l'on se basait majoritairement sur les paramètres issus du réseau lui-même, vers une vue qui serait "centrée utilisateur". Cette vue consiste à considérer l'ensemble des facteurs pouvant impacter la QoE usager. Ces facteurs peuvent venir du réseau, du terminal ou de l'application. Ils peuvent également dépendre du profil de l'utilisateur lui-même, comme illustré dans la Figure (1).

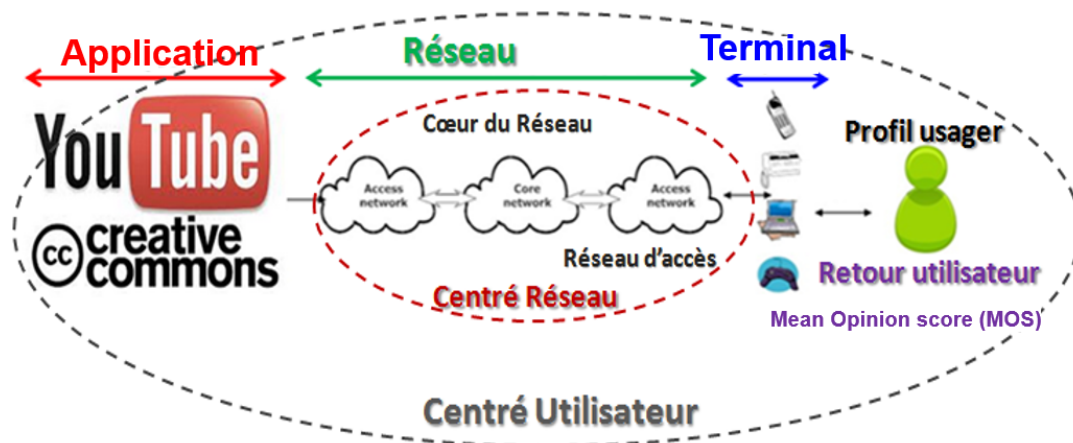


FIGURE 1 – Gestion réseau orienté utilisateur

Nous proposons, à travers les travaux de cette thèse, un nouveau modèle de calcul de la QoE basé sur les méthodes d'apprentissage automatique (*Machine Learning (ML)*). Le modèle proposé est validé dans le cadre d'une nouvelle solution de diffusion vidéo à débit variable (ABR) sensible à la QoE. Nous avons organisé notre travail comme suit :

- i) Construction d'une base de données, la plus complète possible, qui met en exergue un nombre important de facteurs influant la QoE.
- ii) Proposition d'une méthode permettant, en fonction du contexte, de sélectionner les facteurs les plus pertinents qui permettent d'estimer au mieux le ressenti usager.
- iii) Évaluation des méthodes existantes de calcul de la QoE en utilisant les facteurs sélectionnés.
- iv) Proposition d'une nouvelle méthode de calcul de la QoE usager pouvant s'adapter à différents contextes.

La méthode proposée dans le cadre de cette thèse est testée et validée sur un cas d'usage réel, celui de la diffusion vidéo à débit variable (ABR) sensible à la QoE.

Chronologie

Dans cette thèse, nous avons travaillé principalement sur la mise en œuvre d'une méthode de calcul de la QoE usager en utilisant les méthodes issues de l'apprentissage automatique. Parallèlement à cela, nous avons mené des études sur d'autres approches comme les méthodes de régression statistique, les modèles de la logique floue ou les algorithmes génétiques. Le déroulement de cette thèse suit un ordre chronologique décrit ci-dessous :

- Nous nous sommes intéressés au démarrage de la thèse à l'évaluation de la QoE usager par les méthodes subjectives. Nous avons réalisé un état de l'art de ces méthodes. Ensuite, nous avons opté pour l'utilisation de la méthode dite *Note Absolue par Catégorie (ACR : Absolute Category Rating)*[2]. Elle est en effet simple, pratique et facile à mettre en place. Nos travaux se distinguent par le fait d'évaluer la QoE à la fois dans un environnement contrôlé et non-contrôlé (*crowdsourcing*). Ces évaluations nous ont permis de construire 3 bases de données que nous avons présenté dans le cadre des conférences internationales WWIC'2015 [29] et ACM-MSWIW'16 [98] et que nous avons mis en accès libre [93, 95, 96].

- Nous nous sommes ensuite penchés sur la question de l'estimation de la QoE usager. Nous avons réalisé une étude comparative des méthodes de régression statistique et des méthodes d'apprentissage automatique. Celles-ci utilisent en entrée plusieurs facteurs qui impactent la mesure de la QoE usager et calculent au mieux cette mesure. Nous avons apporté une amélioration à ces méthodes en proposant une nouvelle approche qui consiste à sélectionner, en fonction du contexte, les facteurs les plus pertinents à utiliser. Cette approche permet d'optimiser globalement la mesure de la QoE usager. Elle se base sur l'utilisation d'une méthode enveloppante ("*wrapper*") [49] qui intègre le processus de sélection des facteurs et le calcul des valeurs estimées par les méthodes d'apprentissage automatique. Pour cela, nous avons utilisé la corrélation de Pearson [73] ainsi que les méthodes d'apprentissage automatique classiques [113, 32] et ensemblistes [37, 61]. Cela a permis d'améliorer les performances d'estimation de la QoE et de proposer notre contribution [97] à la conférence IEEE-ICC'16.

- Nous avons poursuivi nos investigations par la suite afin d'améliorer l'estimation de la qualité perçue par l'utilisateur. L'idée maîtresse est de ne pas se contenter d'utiliser une seule méthode d'apprentissage automatique pour évaluer la QoE, mais d'en utiliser plusieurs en même temps, afin d'affiner ce calcul. Ainsi, nous avons proposé une nouvelle méthode de mesure de la QoE appelée : *Combined Belief Function (CBF)*. Cette méthode se base sur l'usage des fonctions de croyance et de la fusion d'information en utilisant le modèle des Croyances Transférables (MCT) [131]. Nous avons effectué une évaluation de cette approche en utilisant les données issues de nos campagnes de test [93]. Cela a permis de confirmer l'apport de notre méthode CBF par rapport aux méthodes d'apprentissage automatique et a résulté à la soumission d'un article décrivant notre méthode de calcul de la QoE à une revue internationale.

- Utilisant notre méthode de calcul de la QoE usager, nous avons travaillé sur la proposition d'une solution de diffusion vidéo à débit variable (ABR) compatible avec la norme Dash [66]. Pour ce faire, une synthèse des différentes méthodes utilisées pour construire de telles solutions fut présentée. Ensuite, nous avons détaillé la solution proposée qui se base sur un modèle paramétrique et qui optimise en temps réel la représentation vidéo à afficher en fonction de la

variation des mesures instantanées de la QoE calculées. Le but de cette solution est d'améliorer l'expérience globale de l'utilisateur, tout en optimisant l'utilisation des ressources du réseau et en ajustant en temps réel la qualité de la vidéo envoyée au dispositif d'affichage. Afin d'évaluer cette solution, nous avons développé une plateforme pour la collecte de traces réelles des réseaux traversés dans le cadre de 3 scénarii de test [27]. Les résultats de l'évaluation confirment que la solution proposée améliore les performances des systèmes actuels de diffusion en termes de QoE et d'optimisation de ressources utilisées. Cela a donné lieu à la soumission de cette solution de diffusion vidéo à débit variable à une revue internationale.

Contributions

Le sujet principal de la thèse porte sur le développement de mécanismes adaptatifs capables de récupérer des éléments d'information issus d'un environnement et d'un contexte donnés (Qualité de l'Expérience(QoE)) pour enclencher des actions spécifiques correctrices, en fonction de la perception de l'utilisateur. Les principales contributions de cette thèse sont décrites ci-dessous :

- La première contribution concerne la proposition d'une approche de sélection de facteurs à utiliser pour évaluer la QoE usager. Elle est présentée dans le Chapitre (3). L'idée maîtresse défendue ici consiste à identifier les facteurs les plus pertinents à utiliser lors de l'évaluation de la QoE usager.
- La deuxième contribution quant à elle concerne la présentation d'un nouveau modèle d'estimation de la perception usager (QoE). Ce modèle combine l'utilisation de plusieurs méthodes issues de l'apprentissage automatique en utilisant la théorie des fonctions de croyance et la fusion d'informations. Il est détaillé et validé dans le Chapitre (4).
- La troisième contribution vise à proposer une nouvelle solution de diffusion vidéo à débit variable (*Adaptive Bit Rate (ABR)*), régulé par la QoE. Cette solution se base sur un

modèle paramétrique qui optimise en temps réel la représentation vidéo à afficher en fonction de la mesure instantanée de la QoE estimée. Elle est expliquée et évaluée dans le Chapitre (5).

Organisation de la thèse

Chapitre 1 Ce chapitre introduit dans un premier temps le concept de la QoE et son usage dans le domaine du multimédia. Il réunit les éléments nécessaires de la bibliographie à notre étude. La première partie de ce chapitre porte tout d'abord sur la définition de l'expérience usager (QoE) et son importance pour les acteurs de l'Internet. Dans la deuxième partie, nous passons en revue les méthodes, subjectives et objectives, utilisées pour évaluer la QoE. Enfin, nous terminons par une étude comparative des travaux sur la QoE usager.

Chapitre 2 Ce chapitre est consacré à la présentation des campagnes de test subjectives réalisées dans le cadre de cette thèse. Un état de l'art d'études qui ont élaboré des tests subjectifs a été présenté dans la première partie, tandis que les campagnes de test réalisées durant cette thèse, qui concernent à la fois l'environnement contrôlé (laboratoire) et l'environnement non-contrôlé (*crowdsourcing*), ont été détaillées dans la deuxième partie du chapitre.

Chapitre 3 Ce chapitre présente l'utilisation des méthodes d'apprentissage automatique pour l'évaluation de l'expérience usager. Il introduit, d'une part, une approche pour la sélection des facteurs les plus pertinents à utiliser par ces méthodes pour mieux estimer la QoE usager et détaille, d'autre part, les méthodes d'apprentissage automatique utilisées dans les travaux de cette thèse. Pour finir, nous présentons une évaluation de ces méthodes ainsi que celle de notre approche pour la sélection de facteurs en utilisant l'une des bases de données [93] présentée dans le Chapitre (2).

Chapitre 4 Ce chapitre présente notre proposition d'évaluation de la QoE usager appelé : *Combined Belief Function (CBF)*. Ce modèle se base sur l'utilisation de plusieurs méthodes

d'apprentissage automatique, sur la théorie des fonctions de croyance et également sur la fusion d'information. Une évaluation du modèle a été présentée pour montrer son efficacité comparativement aux méthodes traditionnelles de l'apprentissage automatique.

Chapitre 5 L'objectif de ce chapitre est de mettre en application la méthode de calcul de la QoE proposée (*CBF*) dans le cadre d'une application réelle. Dans ce contexte, nous avons proposé une nouvelle solution de diffusion vidéo à débit variable (ABR) sensible à la QoE. Les performances de la solution proposée sont ensuite évaluées en les comparant à des méthodes de l'état de l'art sur la base de traces réelles.

Ce mémoire de thèse se conclut par un ensemble de pistes et de perspectives pour des recherches futures.

Chapitre 1

État de l'art de l'Expérience utilisateur (QoE)

1.1 Introduction

Ces dernières années ont été le théâtre de changements majeurs dans le monde de l'Internet. D'une part, nous avons assisté à une diversification des périphériques connectés et d'autre part, nous avons assisté à une augmentation et à une cohabitation de nouveaux services de plus en plus demandeurs en ressources comme la vidéo à la demande ultra-HD (UHD), la réalité virtuelle (VR) ou encore les jeux en ligne. Si on tient compte également de l'environnement fortement concurrentiel entre les différents fournisseurs de services Internet[41], on constate que ces changements ont permis d'accroître les exigences et les besoins des usagers, qui souhaiteraient payer ces services à des prix plus bas tout en améliorant leur qualité. Par conséquent, les fournisseurs de services doivent s'adapter à ces nouveaux besoins des utilisateurs. Cela est particulièrement vrai en ce qui concerne le système de diffusion du trafic vidéo qui représente, selon le rapport 2017 de CISCO [50], 75% du trafic sur Internet. Dans cet écosystème, toutes les parties prenantes (fournisseurs de services et opérateurs réseau) utilisent habituellement des politiques de supervision du réseau afin de garantir les niveaux de qualité inscrites dans les contrats d'engagement (*Service Level Agreements (SLA)*). Ces politiques consistent dans la majorité des cas à

surdimensionner les infrastructures en se basant uniquement sur des paramètres quantifiables appelés paramètres de la qualité de service (QoS), issus des couches 2 et 3 du modèle OSI[191]. Toutefois, ces politiques ne sont pas suffisantes. Elles sont complexes et difficiles à mettre en œuvre. De plus, il s'avère que la vision qui est dérivée par ce paradigme ne garantit pas forcément une haute satisfaction des usagers nécessaire pour les fidéliser. Il devient donc intéressant de penser à des paradigmes différents, centrés usager et non uniquement réseau. L'exploration des approches qui se basent sur de nouvelles métriques comme la Qualité d'Expérience (QoE) peuvent y répondre.

Ce premier chapitre introduit la QoE avec comme prisme son utilisation pour les services multimédia. Dans un premier temps, nous déroulons les différentes définitions de la QoE présentes dans la littérature. Dans un second temps, nous nous intéresserons aux méthodes existantes qui permettent d'évaluer l'expérience de l'utilisateur. Nous dresserons enfin un état de l'art des modèles d'estimation de la QoE.

1.2 Motivations

La profusion des services à très forte demande en ressources offertes aux usagers de l'Internet, tels que la vidéo à la demande ultra-HD (UHD), la réalité virtuelle (VR), la voix sur IP (VoIP) et les jeux sur le cloud, impliquent de nouvelles problématiques pour les fournisseurs de services. Parmi ces problématiques, on peut citer :

- (i) La complexité pour faire coexister plusieurs services,
- (ii) La difficulté à assurer une qualité de service satisfaisante au regard d'une infrastructure réseau (et surtout le cœur du réseau) qui peut être inadaptée [169],
- (iii) L'augmentation de l'exigence des usagers.
- (iv) Le développement de la concurrence. 65 services de la vidéo à la demande (VoD) étaient actifs en France à la fin de 2017 [41].

Pour faire face à ces problématiques, les fournisseurs de services utilisaient traditionnellement des politiques de supervision du réseau, qui se basent uniquement sur des paramètres de l'environnement, appelés paramètres de qualité de service (QoS). À cet égard, les métriques

définies par le groupe de travail IETF-IPPM¹ jouent encore aujourd'hui un rôle clé. Ils se concentrent sur la mesure de la qualité, de la performance et de la fiabilité des protocoles et des services qui opèrent sur la couche IP [33]. Les normes et standards définis par l'Union Internationale des Télécommunications (ITU-T : *International Telecommunication Union-Telecommunication Standardization Sector*) jouent également un rôle clé à cet égard. À titre d'exemple, le standard *ITU-T : Y.1540* définit les paramètres utilisés pour évaluer la vitesse, la précision, la fiabilité et la disponibilité des services de transfert de paquets IP. Tandis que le standard *ITU-T : Y.1541* précise les objectifs de performance à atteindre. Sur un autre plan, NetFlow² et IPFIX³ sont des protocoles qui sont souvent utilisés pour capturer les flux dans des points sélectionnés du réseau, appelés mires, afin de mettre en œuvre des systèmes de supervision [69].

Bien que ces politiques sont largement utilisées aujourd'hui, l'expérience a montré que ces mesures ne peuvent pas être utilisées directement pour évaluer le ressenti de l'utilisateur. Ainsi, une prédiction précise du niveau de la qualité réelle (QoE) perçue par l'utilisateur final est nécessaire, car les mesures du niveau 3 au sens du modèle *OSI* ne peuvent pas être utilisées directement pour prédire efficacement le ressenti final de l'utilisateur (QoE) [35]. En outre, une meilleure qualité ne signifie pas forcément une bonne satisfaction de l'utilisateur. Par exemple, il n'est pas profitable de transmettre une vidéo de haute qualité (HD) à un utilisateur qui ne dispose pas d'un terminal ayant un écran à haute définition (HD), car il ne verra pas la différence avec une qualité moindre. Ainsi, le défi qui se pose aux fournisseurs de services est de savoir comment optimiser leurs réseaux tout en assurant la satisfaction des utilisateurs finaux au regard du contexte propre à chaque usager ?

Un début de réponse consiste à faire évoluer la gestion du réseau en migrant d'une vue "centrée réseau" vers une vue "centrée utilisateur". Celle-ci consiste à considérer l'ensemble des facteurs pouvant impacter le ressenti de l'utilisateur final. Ces facteurs peuvent venir du réseau,

1. <https://datatracker.ietf.org/wg/ippm/>

2. <https://www.manageengine.com/products/netflow/>

3. <https://tools.ietf.org/html/rfc7011>

du terminal ou de l'application. Ils peuvent également dépendre du profil de l'utilisateur lui-même. C'est ainsi que les approches développées ces dernières années ne se contentent plus de manipuler des informations de la QoS et des indicateurs clés de performance (*Key Performance Indicator (KPI)*) associés aux SLAs tels que le délai ou la gigue, mais elles intègrent de plus en plus des informations acquises au niveau du périphérique, de l'environnement ou de l'utilisateur telles que les scores subjectifs MOS (*Mean Opinion Score*) [137] ou DMOS (déviation du MOS). L'agrégation de l'ensemble de ces informations est communément appelée expérience de l'utilisateur, plus connue sous l'appellation anglo-saxonne de *Quality of Experience (QoE)*.

1.3 La Qualité d'Expérience (QoE)

Contrairement à la QoS, qui se réfère à des caractéristiques techniques relevées sur les équipements du réseau (délai, gigue, bande passante, etc.), la notion de QoE considère d'autres caractéristiques telles que la perception, les attentes, le comportement, la convivialité, les besoins, le contexte, etc [35]. La QoE n'est pas appelée à remplacer la QoS, mais la complète en fournissant des mesures quantitatives de la perception des utilisateurs des services qui leurs sont proposés [57]. Ainsi, il est possible de compléter les stratégies basées sur la QoS (gestion centrée réseau) en considérant plus de facteurs. Ces facteurs peuvent venir du réseau, du terminal ou de l'application. Ils peuvent également dépendre de la perception et du profil de l'utilisateur lui-même.

Selon la norme ISO 8402, la terme qualité est défini comme "la totalité des caractéristiques d'une entité qui dépend de sa capacité à satisfaire les besoins indiqués et implicites". D'un point de vue historique, nous pouvons observer différentes définitions du terme QoE ou qualité d'expérience. Ce terme a été adopté pour la première fois par Donald Norman en 1986, pour mettre en exergue l'importance de la conception d'un service centrée utilisateur [125]. Il est particulièrement ardu de poser une définition claire de ce qu'est l'expérience utilisateur. Ainsi, nous allons présenter dans la suite quelques définitions qui existent dans la littérature.

La QoE a été définie par l'ITU-T[10] comme "l'acceptabilité globale d'une application ou d'un service, perçu subjectivement par un utilisateur final". Dans cette définition, la mesure de la

QoE est une approche globale de la qualité sur le cycle de vie du service. Cette mesure consiste à évaluer les détails qui peuvent impacter l'ensemble d'une prestation. Comme les utilisateurs finaux dictent le succès ou l'échec de tout service. Les fournisseurs de services et les opérateurs réseau devraient satisfaire leurs besoins en proposant des services performants. Cela permet d'éviter la fuite de leurs utilisateurs et un gain de revenu [23].

Pour l'Institut Européen des Normes de Télécommunications (ETSI), la QoE illustre "une mesure de la performance de l'utilisation d'un service ou d'un produit de la technologie de l'information et de la communication basée à la fois sur des mesures objectives et subjectives"[57].

Pour le groupe QUALINET (*European Network on Quality of Experience in Multimedia Systems and Services*), la QoE est "le degré de satisfaction ou du mécontentement de l'utilisateur d'une application ou d'un service"[102]. Cela exprime la satisfaction au regard de ses attentes en termes d'utilité du service ainsi que de son comportement par rapport à son profil et à son état actuel[15]. Cette définition suggère que la portée de la QoE est passée d'une perspective assez étroite des systèmes de télécommunication à une perspective plus large des services multimédia.

Pour Nokia, la QoE est décrite dans [124] comme "la façon dont un utilisateur perçoit la convivialité d'un service spécifique lorsqu'il l'utilise et qu'il en est satisfait".

Pour Hai, Mellouk et Hoceini [12], la QoE a apporté une amélioration des performances de la gestion de services. En effet, elle représente non seulement les aspects liés à la perception subjective, mais aussi le comportement des utilisateurs et leurs besoins, la pertinence, le contexte, la convivialité et les facteurs humains du contenu livré.

Une autre définition est donnée par Rehman Laghari dans [99], qui définit la QoE comme "un ensemble de facteurs humains centrés sur la base des aspects subjectifs de l'homme. Ceux-ci sont liés à l'interaction d'une personne avec la technologie et avec des entités commerciales dans un contexte particulier".

Au travers de ces différentes définitions de la QoE, nous constatons qu'il n'y a pas de définition complète et générale pour le moment. C'est une notion complexe qui prend en compte de nombreux paramètres, objectifs et subjectifs, dont certains sont liés aux sentiments et à la perception subjective de l'individu.

1.4 Les facteurs d'influence de la QoE

Au cours des dernières années, on a pu observer un changement de paradigmes traditionnels par la prise en compte de l'expérience (QoE) utilisateur dans le contrôle du réseau. La prédiction de la QoE se fait à partir de nombreux paramètres (KPIs), endogènes et exogènes, qui sont issus de l'ensemble des composants de la chaîne de traitement et de transport (utilisateur, périphérique, application, éléments du réseau). La Figure (1.1) présente un aperçu de ces paramètres[35].

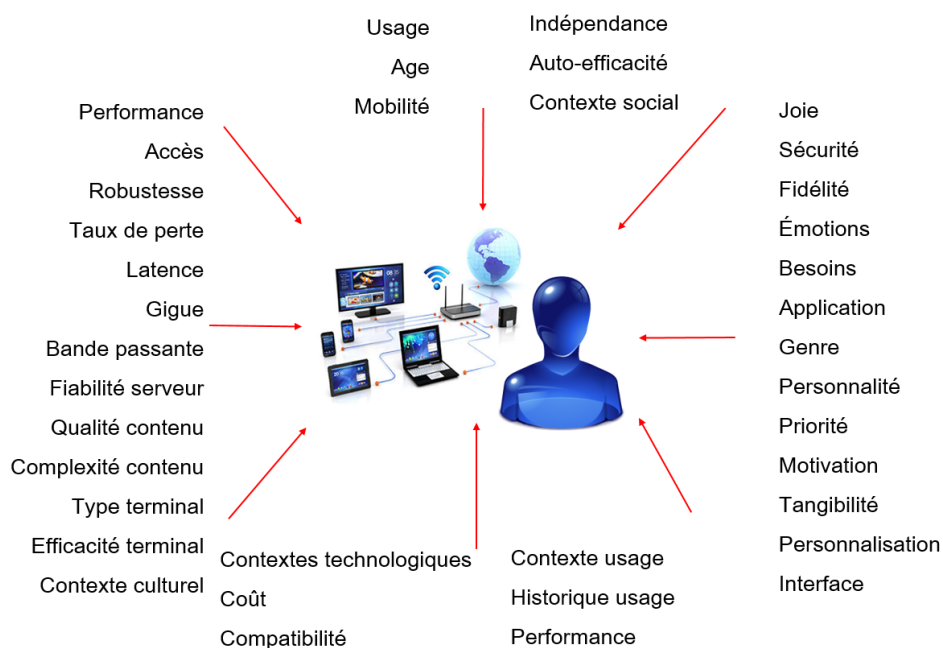


FIGURE 1.1 – Facteurs impactant la QoE

Ickin et al. [148] appelle tous ces paramètres "*facteurs d'influence de la qualité d'expérience (QoE IFs)*" et les définissent comme "toute caractéristique d'un utilisateur, d'un système, d'un service, d'une application ou d'un contexte dont l'état peut avoir une influence sur la qualité d'expérience de l'utilisateur". De nombreux travaux ont proposé de classer ces facteurs. Nous en présentons ci-dessous quelques-uns.

- Ickin et al. [148] ont organisé les facteurs en deux groupes. Le premier concerne les

- paramètres applicatifs et le deuxième le contexte d'utilisation.
- Stankiewicz et Jajszczyk [136] les ont classifié en trois groupes : Celui dépendant de la qualité de service (QoS), celui lié au niveau du service, plus connu sous l'appellation anglaise de *grade of service (GoS)*, et celui de la qualité de résilience, connue sous l'appellation anglaise de *quality of resilience (QoR)*.
 - Callet et al. [102] ont divisé les facteurs en trois catégories, à savoir les facteurs liés à l'humain (*Humain IF*), ceux liés au système (*System IF*) et enfin ceux liés au contexte (*Contexte IF*).
 - Moor et al. [84] ont proposé une architecture modulaire composée de quatre couches : l'entité de surveillance QoS (périphérique, infrastructure, réseau, application); l'entité de surveillance contextuelle (emplacement, mobilité, capteurs, autres applications en cours d'exécution); l'entité de suivi d'expérience (retour usager) et le module utilisateur (règles, expériences).

Dans le cadre de nos travaux, nous avons, quant à nous, proposé une nouvelle classification, publiée dans [29], qui est composée de cinq catégories de facteurs impactant l'expérience usager comme illustré dans la Figure (1.2) ci-dessous.

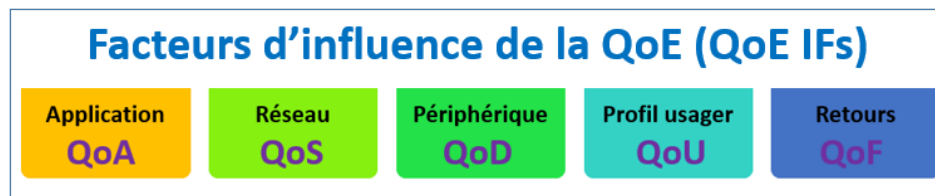


FIGURE 1.2 – Classification des facteurs impactant la QoE en 5 catégories

Où :

- **QoA** : Les paramètres de la couche application. Ils caractérisent le service multimédia (voix, vidéo, jeux en ligne, etc.).
- **QoS** : Les paramètres de la couche réseau. Ils dépendent de l'infrastructure utilisée.
- **QoD** : Les caractéristiques du terminal (information CPU, écran, etc.).
- **QoU** : Les facteurs psychologiques et personnels de l'utilisateur.
- **QoF** : Les informations collectées lors des expériences.

L'objectif de cette classification est de modéliser les facteurs qui impactent l'expérience usager et de les synthétiser sous forme de catégories de facteurs au niveau de l'application, du réseau, du terminal et de l'utilisateur). En se basant sur celles-ci, une analyse discriminante peut être réalisée afin d'identifier l'impact d'une seule catégorie de facteurs sur la QoE. Cette analyse permet d'étudier l'impact de certains facteurs sur l'évaluation de la QoE usager en fonction des facteurs disponibles. Par exemple, les opérateurs réseaux ont les informations sur la bande passante et le délai de bout en bout. Les fournisseurs de services eux disposent des informations sur l'application comme la résolution, le débit binaire vidéo et le nombre d'images par seconde.

Un autre objectif de cette classification est la coopération, entre les fournisseurs de services et les opérateurs réseaux, dans le cas d'un service dont le trafic est chiffré, en leur fournissant une représentation modulaire des facteurs qu'ils peuvent partager entre eux. Par exemple, l'opérateur réseau utilise les informations réseau ("QoS") dont il dispose à son niveau et il les corrèle aux informations de niveau applicatif ("QoA") que le fournisseur de service accepte de les lui fournir. Le protocole QUIC [100] permet cela à travers le champs "PLUS" (*previous "SPUD (Session Protocol for User Datagrams)"*), afin de permettre aux fournisseurs de partager sous la forme de méta-données certains indicateurs de performance avec les opérateurs réseau.

1.5 Évaluation de la QoE

Dans la littérature, il existe une variété d'approches pour évaluer la QoE d'un service multimédia. Ces approches peuvent être classées en deux types selon le degré d'intervention des humains dans les tests. Autrement dit, si la QoE est évaluée directement par les humains, l'approche est dite subjective, sinon l'approche est dite objective. Dans la première approche, les processus d'évaluation utilisent des scores spécifiques [7] comme le score MOS (*Mean Opinion Score*)[137] ou DMOS (dégradation du MOS). La deuxième approche consiste à estimer automatiquement, sans aucune intervention humaine, le niveau QoE via les paramètres/performances du réseau. Nous présentons, dans la suite, chacune de ces approches après avoir introduit quelques recommandations pour la mesure de QoE.

1.5.1 Recommandations

Les organismes de normalisation comme l'ITU et le Groupe des experts de la qualité vidéo *VQEG*⁴ (*Video Quality Experts Group*) ont défini plusieurs méthodologies qui consistent en un ensemble de conditions à respecter pour mener à bien les tests subjectifs. Ces méthodologies couvrent (i) un large éventail de sujets, (ii) une multitude de dispositifs (ordinateurs, tablettes, téléphones portables, etc.) et (iii) un panorama de méthodes d'évaluation. Les tableaux (1.1) et (1.2) décrivent un ensemble des recommandations les plus citées dans la littérature dans les secteurs des radiocommunications (ITU-R) et des télécommunications (ITU-T).

Recommandation	Titre de la recommandation
ITU-R : BT.500-11	Methodology for subjective assessment of the quality of television pictures.
ITU-R : BT.654	Subjective quality of television pictures in relation to the main impairments of the analogue composite television signal.
ITU-R : BT.710-4	Subjective assessment methods for Image quality in high-definition television.
ITU-R : BS.1283	A guide to ITU-R recommendations for subjective assessment of sound quality.
ITU-R : BT.1210	Test materials to be used in subjective assessment.
ITU-R : BT.1788	Methods for subjective assessment of video quality in multimedia applications.

TABLE 1.1 – Quelques recommandations proposées par ITU-R

En pratique, lors de la réalisation des tests subjectifs pour un service multimédia (télévision, vidéo streaming, jeux en ligne, etc.), quelques points notables doivent être pris en considération comme la génération des stimulus (caractéristiques de la séquence de visionnement) ou encore les types utilisés d'échelles par le testeur pour évaluer la qualité d'un stimulus. Pour répondre à ces points, un panorama de méthodes, dits subjectives, est utilisé. L'une des recommandations les plus pertinentes pour l'évaluation subjective de la qualité visuelle est la recommandation "BT.500 – 11" [4] qui est destinée aux signaux de télévision. À noter que la recommandation "P.911" [3] est destinée quant à elle à l'évaluation subjective des services multimédia. Ces

4. <https://www.its.bldrdoc.gov/vqeg/>

Recommandation	Titre de la recommandation
ITU-T : P.10/G.100	Vocabulary for performance quality of service.
ITU-T : P.85	A method for subjective performance assessment of the quality of speech voice output devices.
ITU-T : P.800	Methods for subjective determination of transmission quality.
ITU-T : P.800.1	Mean Opinion Score (MOS) terminology.
ITU-T : P.830	Subjective performance assessment of telephone-band and wideband digital codecs.
ITU-T : P.910	Subjective video quality assessment methods for multimedia applications.
ITU-T : P.911	Subjective audiovisual quality assessment methods for multimedia applications.
ITU-T : P.920	Interactive test methods for audiovisual communications.
ITU-T : P.930	Principles of a reference impairment system for video.

TABLE 1.2 – Quelques recommandations proposées par ITU-T

recommandations offrent des conseils sur les conditions d’observation, la référence pour les observateurs, la sélection du matériel de test et la procédure d’évaluation.

1.5.2 Évaluation subjective

Une méthode subjective d’évaluation de la QoE consiste à collecter des mesures au travers d’expériences (contrôlées ou non contrôlées) en utilisant des scores spécifiques. Elle est réalisée à travers des tests réels avec la collaboration de participants humains. Ceux-ci peuvent être impliqués de manière passive (en regardant/écoutant) ou d’une manière active/interactive (en participant à une conversation) dans les tests. L’objectif est d’avoir un grand échantillon d’observations qui reflète le ressenti de ces participants. Par la suite, ces observations seront utilisées pour construire des modèles de calcul de la QoE usager.

Lors de l’évaluation subjective de la QoE, les participants jugent la qualité de la présentation de certains services proposés (par exemple : la qualité conversationnelle d’un service téléphonique, la qualité d’une vidéo, etc.). Ces tests doivent être soigneusement conçus à l’avance et l’échantillon de participants doit être correctement sélectionné en fonction des directives et des

recommandations validées par les organismes de normalisation [104]. En outre, la métrique la plus connue pour évaluer la QoE dans les tests subjectifs est l'utilisation de la "note d'opinion moyenne", connue sous l'appellation anglaise : *Mean Opinion Score (MOS)* [137]. Celle-ci est une métrique utilisée pour l'évaluation des applications audiovisuelles. Elle est absolue, à savoir qu'elle n'a pas de valeurs négatives dans son échelle. Celle-ci est de 1 à 5, où la note de 5 représente une excellente qualité et la note de 1 représente une très mauvaise qualité [164, 2].

A) Méthodes subjectives d'évaluation des signaux de télévision

A.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 une autre de référence. Les deux séquences sont toujours affichées ensemble. Les testeurs donnent un score selon une échelle de dépréciation : "imperceptible", "sensible mais pas ennuyant", "un peu ennuyant", "ennuyant", et "très ennuyant". Cette échelle peut également être notée sur 5 niveaux, où la note de 5 représente "imperceptible" et la note de 1 indique "très ennuyant". La Figure (1.3) présente une description de cette méthode avec les deux échelles d'évaluation qui peuvent être utilisées.

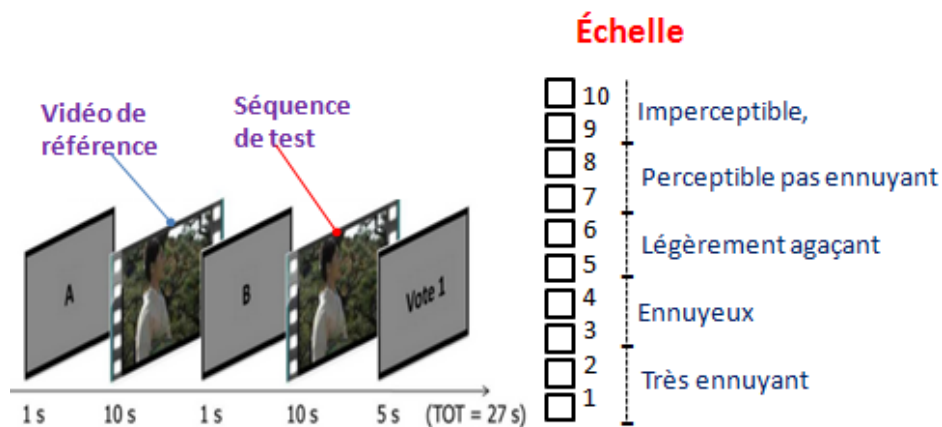


FIGURE 1.3 – Méthode d'évaluation DSIS

A.2) Évaluation de la qualité continue en stimulus unique (SSCQE) :

La méthode SSCQE (*Single Stimulus Continuous Quality Evaluation*) a été conçue à

l'origine pour effectuer des évaluations efficaces, sur une longue durée, de la qualité subjective des services multimédia, dans des conditions proches de l'environnement domestique. Elle consiste à regarder une séquence vidéo de 20 à 30 minutes, sans aucune référence à la séquence originale, et à évaluer la qualité subjective de la vidéo à chaque changement de qualité. Les testeurs donnent un score selon une échelle de dépréciation en utilisant une mesure qui va de "très mauvaise" à "excellente". Cela correspond à une échelle numérique de 0 à 100. La recommandation "ITU-R BT.500-13" donne une description détaillée de cette méthode et la Figure (1.4) présente le déroulement d'un test.

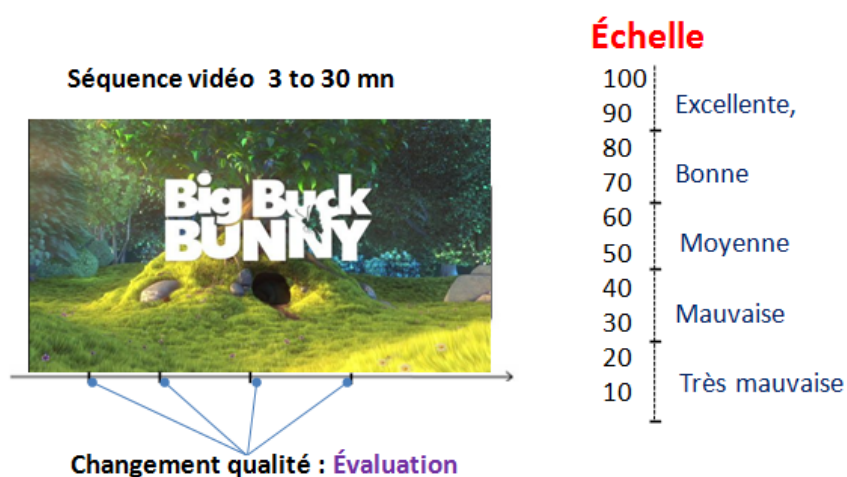


FIGURE 1.4 – Méthode d'évaluation SSCQE

A.3) Échelle continue de la qualité en double stimulus (DSCQS) :

La méthode DSCQS (*Double Stimulus Continuous Quality Scale*) est utile dans le cas où il est difficile de simuler des conditions de test qui représentent les différentes gammes de la qualité. De même que la méthode DSIS, les séquences vidéo sont présentées en paires, une séquence dégradée et l'autre de référence. Mais à la différence du DSIS, les testeurs sont invités à évaluer la qualité des deux séquences et non pas uniquement la séquence dégradée. Dans cette méthode, les séquences vidéo sont de très courte durée (soit 10 secondes) et la qualité est

mesurée sur une échelle continue allant de "très mauvais" à "excellente". Ceci est équivalent à une échelle numérique de 0 à 100 comme indiqué comme la Figure (1.5).

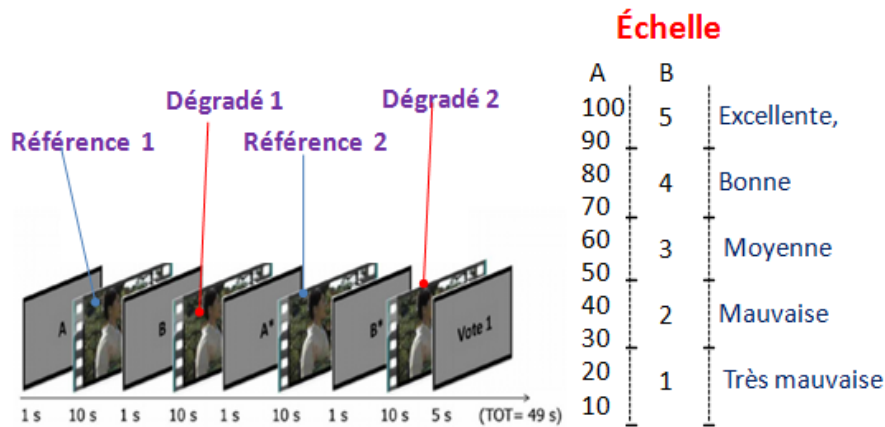


FIGURE 1.5 – Méthode d'évaluation DSCQS

B) Méthodes subjectives d'évaluation des services multimédia

Les méthodes présentées dans la partie précédente sont destinées aux signaux de télévision. Par ailleurs, d'autres méthodes basées sur de légères modifications et certaines adaptations ont été décrites dans la recommandation "ITU-T Rec. P.911" [3], pour l'évaluation subjective de l'ensemble des services multimédia. Dans cette partie, nous présentons les différentes méthodes.

B.1) Note Absolue par Catégorie (*Absolute Category Rating (ACR)*) :

La méthode "Note Absolue par Catégorie", connue sous l'appellation anglaise : *Absolute Category Rating (ACR)*, est la plus couramment utilisée dans l'évaluation des services multimédia [2]. Cette méthode consiste à évaluer la séquence vidéo sans regarder une vidéo de référence. Après la fin de chaque séquence, les testeurs sont invités à donner une évaluation selon l'échelle MOS[137].

B.2) Note par catégories de dégradation (*Degradation Category Rating(DCR)*) :

La méthode "note par catégories de dégradation" (DCR) consiste à présenter des séquences vidéo en paires, une séquence originale (sans aucune dégradation) et l'autre dégradée. La première séquence (vidéo originale) est toujours présentée dans le premier échantillon de chaque paire. La seconde concerne la même vidéo, mais elle est modifiée par les conditions de tests à définir au préalable. Dans cette méthode, les utilisateurs sont invités à évaluer la déformation de la deuxième séquence par rapport à l'originale. L'échelle de notation utilisée a 5 niveaux allant de l'imperceptible (5) à très ennuyeux (1). La méthode DCR est plus performante que la méthode ACR lorsque la différence de qualité est minime entre les deux séquences vidéos.

B.3) Méthode de comparaison par paires (*paired comparison (PC)*) :

La méthode comparaison par paires (PC) consiste à présenter des paires de séquences vidéo aux testeurs. Ceux-ci estiment la différence de la qualité entre chaque paire de séquences. Toutes les paires sont présentées dans deux ordres, le premier consiste à commencer par la séquence 1 tandis que dans le deuxième, on commence par la séquence 2. Les testeurs doivent seulement exprimer leur préférence entre les deux séquences (au lieu d'attribuer un score à chacune), tout en ne dépassant pas une durée de vote (temps pour exprimer leur préférence) égale à 10 secondes.

Le Tableau (Figure (1.6)) ci-dessous présente une synthèse [101, 3] qui compare les méthodes subjectives présentées dans cette partie. En outre, les faiblesses ainsi que les avantages de chaque méthode sont représentés.

Méthodes subjectives d'évaluation						
	Avec référence			Sans référence		
Méthode	DCR	DSIS	DSCQS	ACR	PC	SSCQE
Service	Multimédia	Télévision	Multimédia	Multimédia	Multimédia	Multimédia
Durée	[8-10 s]	[8-10 s]	[8-10 s]	[8-10 s]	[8-10 s]	[3-30 m]
Stimulus	Paire: - référence - dégradée	Paire: - référence - dégradée	2 stimulus à tetsar	Unique	2 stimulus à tetsar	Unique
Échelle	Catégorielle	Catégorielle	Continue	Unique	/	Catégorielle
Valeurs de l'échelle	-Imperceptible -Pas ennuyeux -Peu ennuyeux -Ennuyeux -Très ennuyeux	-Imperceptible -Pas ennuyeux -Peu ennuyeux -Ennuyeux -Très ennuyeux	0 - 100	-Excellente -Bonne -Moyen -Mauvaise -Très mauvaise	Préférence pour une 2 séquences	1 - 100
Avantages	- Pour des systèmes haute qualité - Meilleure détection de la dégradation	- Meilleure détection de la dégradation	- Étude raffinée de la dégradation	- Quantification Facile - Application rapide	- Utile pour comparer 2 systèmes quasi-égales	- Étude des fluctuations de la qualité en temps réel
Faiblesses	- Séquence de test de durée de 10 secondes - Évaluation réalisée en posteriori					
	- Évaluation de la qualité explicite - Biais de représentativité - Biais de mesure					

FIGURE 1.6 – Synthèse des méthodes subjectives d'évaluation de la QoE

1.6 Modèles d'estimation de la QoE

Les modèles d'estimation de la QoE visent à calculer le ressenti des usagers finaux à partir des facteurs. Le calcul peut être fait soit par des modèles objectifs, soit par des modèles d'estimation hybrides qui utilisent des données réelles d'utilisateurs pour entraîner des méthodes de régression statistique ou des méthodes d'apprentissage automatique.

1.6.1 Modèles objectifs

Les modèles objectifs de la QoE fournissent automatiquement une estimation de qualité basée sur la valeur de certaines mesures en entrée, un peu comme si nous demandions à un observateur humain d'évaluer un stimulus donné (par exemple, un clip vidéo de dix secondes)

[176]. Autrement dit, ces méthodes sont des techniques instrumentales qui produisent, à partir de plusieurs mesures, des résultats qui se rapprochent des scores qui seraient obtenues en utilisant des méthodes subjectives.

Dans la littérature, les méthodes objectives de l'évaluation de la qualité vidéo peuvent être classées en trois catégories, selon qu'elles utilisent des informations extraites de la vidéo origine (c.f. Figure (1.7)) : les méthodes avec référence complète, les méthodes avec référence réduite et les méthodes sans référence.

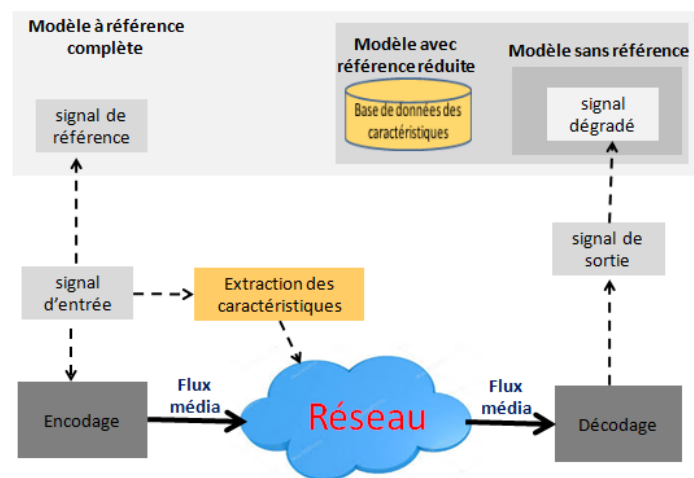


FIGURE 1.7 – Modèles objectifs selon les informations utilisées

Dans le cas des méthodes avec référence complète, connues sous l'appellation : *Full Reference (FR)*, toutes les informations de l'image/vidéo originale sont disponibles. En conséquence, les méthodes *FR* sont basées sur la comparaison d'une image ou d'une vidéo déformée avec l'image ou la vidéo d'origine. Les recommandations "J.144", "J.247", et "J.341" de l'ITU-T décrivent ces méthodes.

Dans le cas des méthodes avec référence réduite, connues sous l'appellation : *Reduced Reference (RR)*, il n'est pas nécessaire de donner accès à l'image ou à la vidéo originale, mais seulement de fournir quelques caractéristiques représentatives de la texture de l'image ou de la vidéo originale. La comparaison de l'information réduite de l'image ou

de la vidéo originale avec l'information correspondante de l'image ou de la vidéo déformée fournit l'entrée pour les méthodes RR. Les recommandations "J.246" et "J.342" de l'ITU-T détaillent ces méthodes.

Dans le cas des méthodes sans référence (*No Reference(NR)*), aucune information de l'image ou de la vidéo originale n'est utilisée. Cette classe de méthodes recherche les artefacts par rapport aux caractéristiques des pixels d'une image ou d'une vidéo, ou bien, elle utilise l'information intégrée dans le flux binaire du format image ou vidéo. Les méthodes NR peuvent être classées en trois types en fonction du niveau d'information traitée. Il s'agit des méthodes basées sur les pixels, ("NR-P"), les méthodes basées sur les flux binaires ("NR-B") et les méthodes hybrides qui combinent les deux premières "NR-P" et "NR-B". La Figure (1.8) présente ces méthodes.

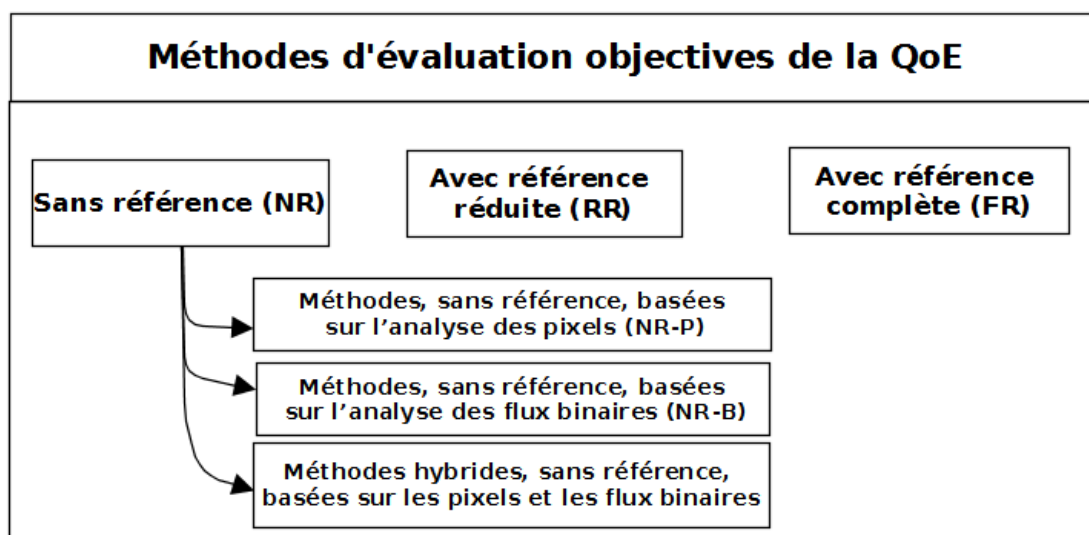


FIGURE 1.8 – Classification des méthodes objectives

1.6.2 Métriques de l'évaluation de la qualité vidéo

Différentes métriques sont utilisées pour estimer la qualité de la vidéo. Les plus populaires sont le rapport signal/bruit et l'erreur quadratique moyenne. Ces mesures sont calculées par les méthodes FR et elles sont basées sur une comparaison pixel par pixel. Il existe également

des mesures telles que la valeur de la "similarité structurelle", la métrique VQM de la NTIA⁵ et le score VMAF de Netflix⁶. Ces métriques utilisent un certain nombre de paramètres de l'image/vidéo tels que le flou, le blocage, la luminosité ou le degré de mouvement pour estimer la qualité vidéo. Dans cette partie, nous présentons les métriques les plus utilisées pour l'évaluation objective de la QoE

Le rapport de "Signal/Bruit" (*Peak Signal to Noise Ratio(PSNR)*) est le paramètre de mesure objective qui calcule la différence mathématique entre chaque pixel de la vidéo codée avec celui de la vidéo originale [99]. L'erreur quadratique moyenne (*Mean Squared Error (MSE)*), est aussi un paramètre très utilisé pour l'évaluation de la qualité des techniques vidéo. La popularité de ces deux métriques (PSNR (Équation (1.1)) et MSE Équations (1.2)) est liée à la simplicité de leurs formules et au faible temps de calcul dont elles ont besoin.

$$PSNR(dB) = 20 \times \log\left(\frac{255^2}{\sqrt{MSE}}\right) \quad (1.1)$$

$$MSE = \frac{\sum_{j=1}^N \left(\sum_{i=1}^M (X_{i,j} - Y_{i,j})^2 \right)}{M \times N} \quad (1.2)$$

Où : $X_{i,j}$ signal original, $Y_{i,j}$ signal reçu, $M \times N$ taille de l'image. 255 représente la valeur de luminosité maximale. Une valeur moyenne de PSNR allant de 30 à 40 caractérise une vidéo de haute qualité.

La similarité structurelle (*Structural Similarity Index-SSIM*-) est une autre métrique du mode avec référence (FR). Elle compare les informations en utilisant la luminosité, le contraste et la similarité structurelle entre l'image originale et l'image traitée [178]. Les valeurs acceptables de cette métrique sont de "-1" (différence maximale) à "1" (aucune différence). Une valeur supérieure signifie une meilleure qualité. L'équation (1.3) donne la mesure en utilisant SSIM

5. <https://www.ntia.doc.gov/>

6. <https://www.netflix.com/>

entre deux fenêtres d'images x et y de taille commune $N \times N$:

$$SSIM(x, y) = \frac{(2u_x u_y + c_1)(2\sigma_{xy} + c_2)}{(u_x^2 + u_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \quad (1.3)$$

u , σ_x^2 et σ_{xy} sont la moyenne, la variance et la covariance respectivement. c_1 et c_2 sont deux variables constantes.

Une autre méthode objective est appelée *Video Quality Metric* (VQM) [112]. Cette méthode a été initiée par le NTIA⁷ (*Administration Nationale des Télécommunications et de l'Information*). Elle utilise les routines de calibrage liées aux caractéristiques des images pour l'estimation de la qualité de la vidéo. Les techniques de calibrage ont été associées à ce modèle par le groupe de recherche VQEG⁸ afin d'évaluer indépendamment la perception de l'utilisateur du service vidéo. En fait, les coefficients de ce modèle dépendent principalement de la complexité au regard du contenu de l'information de la séquence analysée. Le modèle VQM proposé est défini comme suit :

$$\begin{aligned} VQM = & 0.2097 * si_{loss} + 0.5969 * hv_{loss} + 0.2483 * hv_{gain} + 0.0192 * chroma_{spread} \\ & - 2.3416 * si_{gain} + 0.0431 * si_{atigai} + 0.0076 * chroma_{extreme} \end{aligned} \quad (1.4)$$

avec :

- si_{loss} : Paramètre qui représente la perte d'information spatiale (similitude entre les pixels adjacents), comme la perte dans l'information de bord de la séquence vidéo.
- hv_{loss} : Paramètre qui représente la perte relative d'informations spatiales dans les directions horizontales et verticales par rapport à celles situées dans les directions diagonales.
- hv_{gain} : Paramètre qui représente l'augmentation involontaire des informations spatiales dans les directions horizontales et verticales en raison, par exemple, de blocage des artefacts.

7. <https://www.ntia.doc.gov/>

8. <https://www.its.bldrdoc.gov/vqeg/vqeg-home.aspx>

- $chroma_{spread}$: Détecte les changements dans la distribution des couleurs.
- si_{gain} : Le gain dans l'information spatiale de la vidéo reconstruite qui pourrait résulter des mesures d'affinage des arêtes utilisées par le décodeur vidéo.
- $si_{ati_{gai}}$: Paramètre qui explique l'effet de masquage des déficiences temporelles s'il existe beaucoup d'activités spatiales dans la scène, ce qui les rend moins perceptibles. Un effet de masquage similaire des déficiences spatiales lorsqu'il y a beaucoup de mouvements dans la scène s'applique également et est pris en compte.
- $chroma_{extreme}$: Détecte une distorsion brute localisée dans l'espace de couleur.

Le résultat de la valeur "VQM" est interprétée comme suit : 0 signifie aucune différence (meilleure qualité), plus la valeur est élevée, plus la différence est grande (moins bonne qualité).

Un autre exemple récent est la méthode dite "VMAF" (*Video Multi-Method Assessment Fusion*) utilisé par Netflix [24]. L'objectif de cette méthode est d'estimer la QoE usager en combinant plusieurs métriques élémentaires de qualité vidéo. Le raisonnement de base est que chaque mesure élémentaire peut avoir ses propres impacts en ce qui concerne les caractéristiques du contenu source, le type d'artefacts et le degré de distorsion. Les métriques élémentaires sont fusionnées afin de calculer une métrique finale. Cela est réalisé à l'aide d'un algorithme d'apprentissage automatique qui est le modèle à base des machines à vecteurs de support(SVM). Il fournit une métrique finale (score VMAF) allant de 0 à 100, où une valeur grande (proche de 100) signifie une meilleure qualité.

1.6.3 Modèles hybrides

L'objectif de cette section est de présenter un état de l'art des méthodes hybrides d'estimation de la QoE. Ces méthodes ont pour but d'estimer la QoE des utilisateurs en utilisant des données collectées lors des tests subjectifs. Pour cela, elles utilisent en entrée un ensemble de facteurs et elles calculent des scores MOS prédits en utilisant, soit des méthodes de la régression statistique, soit des méthodes de l'apprentissage automatique. La Figure (1.9) présente une description de ces méthodes hybrides d'estimation de la QoE.

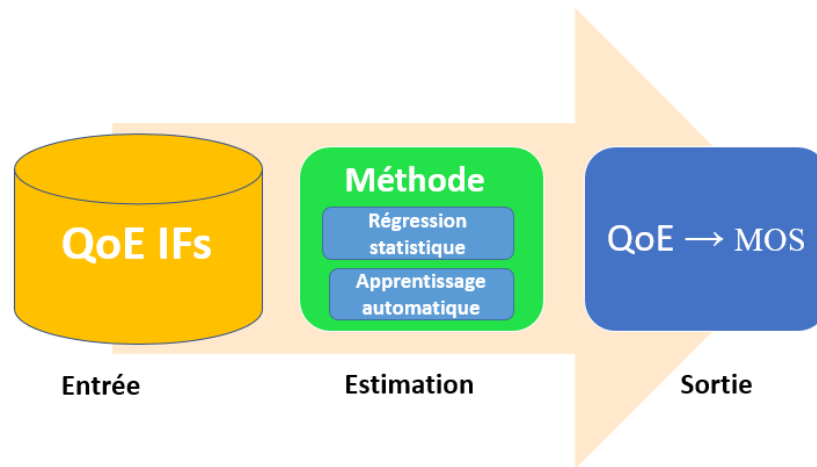


FIGURE 1.9 – Méthodes hybrides d’estimation de la QoE

1) Modèles hybrides de la QoE basés sur les régressions statistiques

Du point de vue mathématique, la relation (corrélation) entre la QoE et les facteurs mesurés a été simplement décrite sous une forme linéaire, logarithmique, exponentielle ou fractionnaire. Le tableau ci-dessous (Tableau (1.3)) décrit quelques modèles mathématiques qui peuvent relier la QoE et les facteurs qui l’impactent, où la QoE présente la qualité d’expérience usager, les f_i et f_j sont les facteurs qui l’impactent et les $a_i, c_k, \alpha, \alpha_i, \beta$ et γ représentent les coefficients des modèles utilisés.

Nom	Linéarité	Forme	Relation
Linéaire	Oui	Linéaire	$QoE = a_1 * f_1 + a_2 * f_2 + \dots + a_n * f_n$
Loi Weber-Fechner [130]	Non	Logarithmique	$QoE = a_i * \ln(f_j)$
Hypothèse IQX [110]	Non	Exponentielle	$QoE = \alpha * \exp^{-\beta * f_j} + \gamma$
Loi Stevens [151]	Non	Puissance	$QoE = a_i * f_j^{b_k} + c_k$
Fractionnaire [63]	Non	Fractionnaire	$QoE = \frac{a_i * f_i}{a_j * f_j}$

TABLE 1.3 – Modèles mathématiques représentant la relation entre la QoE et les facteurs

Linéaire

La relation entre l'estimation de la valeur QoE et les facteurs qui l'impactent peut être considérée sous une forme linéaire ou non linéaire [26]. Korhonen et al.[76] ont exprimé cette relation sous une forme linéaire. Ils ont utilisé la qualité vidéo (VQ) pour calculer la QoE en termes de score MOS. Le modèle présenté est illustré dans l'équation (1.5).

$$MOS_{video} = a_1 + a_2 * VQ \quad (1.5)$$

Pour trouver les paramètres a_1 et a_2 , un calcul de l'ajustement linéaire entre la qualité vidéo VQ générée et le retour usager (en termes de scores MOS calculés) est réalisé.

Logarithmique

Dans [130], Reichl et al. ont montré que la relation entre la valeur de la QoE estimée et les facteurs qui l'impactent peut être de nature logarithmique dans le cas d'un service web. Pour arriver à cette conclusion, les auteurs ont réalisé une campagne de test subjective avec 85 personnes pour évaluer plusieurs services web sur des périphériques mobiles comme la navigation, le courrier électronique et le téléchargement. Pour ce dernier, le modèle de calcul de la QoE est exprimé, en fonction de la taille du fichier s et de la bande passante réseau bdw , comme illustré dans l'équation (1.6) :

$$QoE_{telechargement} = \frac{0.775}{\sqrt{s}} * \ln(bdw) + 1.268 \quad (1.6)$$

Concernant la navigation web, Khirman et al.[149] ont proposé une relation non linéaire sous forme logarithmique, illustrée par l'équation (1.7) ci-dessous.

$$QoE_{web}(bdw) = -0.017 * \log(bdw) + 0.1 \quad (1.7)$$

Chen et al.[144] ont présenté une modélisation sous une forme logarithmique pour évaluer le service voix sur IP (VoIP). Dans cette étude, les auteurs visent à étudier les effets de la bande passante réseau. Une campagne de test qui concerne le service VoIP (Skype avec le codec SILK[9])

a été effectuée avec un panel de 127 participants pour obtenir leur ressenti en termes de score MOS. Grâce aux résultats de l'expérimentation, les auteurs ont montré que la variation de la bande passante affecte la QoE usager sur une échelle logarithmique en suivant la loi de Weber-Fechner (Tableau (1.3)). La relation entre la valeur de la QoE estimée des utilisateurs de la VoIP et le débit binaire (br) est présentée ci-dessous (équation 1.8) :

$$QoE_{VoIP} = \gamma + \ln(br + \alpha) + \beta \quad (1.8)$$

Où : br est le débit binaire audio, α , β et γ sont des coefficients à déterminer en fonction des caractéristiques du codec SILK (nouveau codec de Skype)[9].

Fractionnaire

Dans cette partie, nous présentons les modèles d'estimation de la QoE qui se basent sur des fonctions fractionnaires. Kim et al. [63] ont défini un modèle qui prédit la QoE usager en termes de scores MOS pour différents services multimédia (VoIP, vidéo à la demande, etc.). Le modèle tient compte à la fois de plusieurs critères de qualité de service (QoS) ainsi que des caractéristiques du service multimédia. Il est exprimé mathématiquement par l'équation (1.9).

$$QoE = k \left\{ \frac{e^{QoS-\alpha} + e^{-QoS+\alpha}}{e^{QoS-\alpha} + e^{-QoS+\alpha} + \beta} + 1 \right\} \quad (1.9)$$

Où :

- α représente la classe de la "QoS" du niveau réseau.
- β est la constante du niveau du service, déterminée en fonction de la classe de ce service.
- k généralise la qualité subjective du service utilisé et est calculé expérimentalement.
- QoS est le score de la qualité de service normalisé (équation 1.10).

$$QoS = f(D, E, J, L, E) \quad (1.10)$$

Où : D représente le retard, J la gigue, L la perte de paquets, E les erreurs de paquets,
 B c'est la bande passante et f est la fonction qui calcule le score QoS normalisé.

Un autre modèle d'estimation fractionnaire a été proposé par Wang et al. dans [166]. Dans cette étude, les auteurs se sont inspirés de la méthode objective VQM (Section (1.6.1)) pour proposer une nouvelle méthode non intrusive, qui calcule la QoE usager au niveau du périphérique

du service streaming vidéo. Les valeurs des métriques du modèle VQM sont présentées dans le Tableau (1.4). L'équation (1.11) ci-dessous représente le poids des paramètres utilisés :

$$VQM = \frac{P_1 + P_2 * X_1 + P_3 * X_1^2 + P_4 * X_2 + P_5 * X_2^2}{1 + P_6 * X_1 + P_7 * X_1^2 + P_8 * X_2 + P_9 * X_2^2} \quad (1.11)$$

Avec : $0 \leq X_1 \leq 5$, et $0 \leq X_2 \leq 5$, X_1 représente la gigue en termes de millisecondes (*ms*), et X_2 représente le taux de perte de paquets.

Constante	Valeur
P_1	0.00421683208981546
P_2	-0.00173344507412777
P_3	0.0935525137413507
P_4	0.788253736441116
P_5	-0.113184563779598
P_6	0.140708756966682
P_7	0.064602569193934
P_8	0.814736390397921
P_9	-0.147553823779441

TABLE 1.4 – Valeurs des coefficients du modèle fractionnaire proposé par Wang et al.

Ce modèle de Wang et al.[166] a été comparé au modèle général VQM (Section 1.6.1). L'erreur entre les deux modèles varie en fonction du contenu de la vidéo (degrés de mouvement dans la séquence vidéo). Pour améliorer leur modèle, les auteurs ont proposé d'établir une étude pour analyser l'impact de contenu vidéo (degré de mouvements) sur la QoE usager.

Puissance

Khorsandroo et al. ont utilisé dans [150] une relation sous la forme de puissance pour calculer la qualité d'un service de streaming vidéo. Les auteurs ont réalisé un test subjectif pour récolter une base de données de 440 exemples en variant le taux de perte des paquets (*loss*). En utilisant la base de données récoltée, une relation non-linéaire sous forme de puissance a été construite pour modéliser la relation entre le taux de perte de paquets et la QoE des usagers du

streaming vidéo. L'équation (1.12) décrit la relation trouvée.

$$QoE_{video}(loss) = \frac{-0.549}{0.341} * loss^{0.341} + 5.245 \quad (1.12)$$

Cette relation permet a donné une corrélation égale à 0.989 et un taux d'erreur carrée moyenne (MSE) inférieur à 0.01.

Par la suite, Khorsandroo et al. [151] ont réalisé une comparaison du modèle puissance par rapport à la loi Weber-Fechner (relation logarithmique). Dans cette étude, un test subjectif a été mené en utilisant le lecteur du flux vidéo VLC pour construire une base de données subjective. Celle-ci contient le taux de perte de paquets (*loss*) et le score MOS usager. En utilisant cette base de données, les auteurs ont construit et comparé deux modèles : logarithmique et puissance. La Figure ci-dessous (1.10) nous décrit les relations et les performances des 2 modèles.

Fonction	Modèle de régression	Erreur estimée du modèle	Performance		
			r	R ²	MSE
Puissance	$QoE = -0.327(QoS)^b + 3.948$	0.068	0.966	0.934	0.005
Logarithmique	$QoE = -0.844 \ln(QoS) + 3.753$	0.122	0.983	0.967	0.015

FIGURE 1.10 – Résultats des comparaisons entre les modèles puissance et logarithmique

Les auteurs ont expliqué qu'en raison du comportement imprévisible du trafic réseau dans un réseau informatique réel et du fait que plusieurs paramètres peuvent affecter la QoE en même temps, il est possible que, dans certaines situations, les fonctions de puissance puissent évaluer la QoE mieux que les fonctions logarithmiques.

Exponentiel

Dans cette partie, nous présentons la relation entre la QoE et les facteurs qui l'impactent sous la forme d'une relation exponentielle. Cette approche est appelée aussi : Hypothèse IQX ou plus connu sous le nom anglaise : *IQX hypothesis (exponential interdependency of quality of experience and quality of service)*. Selon Wael [48], "l'hypothèse IQX vise à décrire la qualité perçue

par l'utilisateur (QoE) en surveillant une dégradation de la QoS par une fonction exponentielle telle que présentée par l'équation (1.13) :

$$QoE = \alpha * exp^{-\beta * QoS} + \gamma \quad (1.13)$$

Où : α , β et γ sont des paramètres qui donnent un degré de relation entre la QoS mesurée et la QoE calculée.

Sur la base de cette relation exponentielle, plusieurs études ont été proposées dans la littérature pour proposer des modèles de calcul de la QoE usager pour les différents services de l'Internet.

La première étude utilisant l'hypothèse IQX a été introduite par Fielder et al. dans [110]. Les buts étaient (i) d'étudier l'impact de la perte de paquets, de la gigue et du réordonnement de paquets sur la VoIP, et (ii) de quantifier l'impact du temps de réponse sur la qualité de la navigation web. Des expériences se sont déroulées dans un environnement contrôlé à l'aide de l'émulateur de réseau "NIST Net"⁹. Le modèle IQX a été utilisé ensuite pour construire le module d'estimation de la QoE pour différents services comme le service de la voix sur IP (VoIP) et la navigation Web.

Dans le cas de la VoIP, plusieurs tests ont été menés pour chaque paramètre réseau (la perte de paquets, la gigue et le réordonnement de paquets). Le flux audio perturbé a été capturé à partir de l'application audio "SJPhone"¹⁰ et les utilisateurs ont été invités à évaluer la qualité perçue en fonction de l'échelle MOS. Dans le cas de la VoIP, l'hypothèse IQX donne une métrique de référence réduite $RR(B, X)$, où X représente la distorsion de la QoS qui est dérivée de la comparaison de la quantité de données reçues avec la quantité de données envoyées à partir de l'utilisateur B . Les auteurs ont fait varier la probabilité de perte de paquets (P_{loss}) de 0 à 90% (avec un pas de 0.09%). À la fin, le modèle obtenu pour le codec voix iLBC est présenté ci-dessous dans l'équation (1.14) :

$$QoE_{VoIP} = 3.010 * exp^{-4.473} * P_{loss} + 1.065 \quad (1.14)$$

9. <https://www-x.antd.nist.gov/nistnet/>

10. <https://www.voip-info.org/sjphone/>

Où : P_{loss} présente le taux de perte de paquets.

Dans le cas de la navigation web, les temps de réponse et de téléchargement ont été considérés pour calculer la QoE usager avec l'hypothèse IQX. Les auteurs ont réalisé une campagne subjective de test en utilisant la méthode ACR[2]. Au total, 49 expériences ont été menées pour chaque session. Le modèle d'estimation obtenu pour prédire la QoE de la navigation web est présenté ci-dessous dans l'équation (1.15) :

$$QoE_{Web} = 4.298 * exp^{(-0.347 * S_{time})} + 1.390 \quad (1.15)$$

Où : S_{time} présente le temps d'une session.

Par la suite, Hossfeld et al. ont proposé, dans l'étude [70], un nouveau modèle d'estimation de la QoE usager en utilisant l'hypothèse IQX et en se basant sur des données subjectives. Le but de cette étude est de comparer deux codecs de la voix : le codec *iLBC* (par défaut) et le codec *G711* (norme de compression audio ITU-T). Dans l'expérimentation, les paramètres de la QoS (gigue, perte de paquets (P_L)) et le taux de réordonnancement (ρ) ont été testés. Pour l'émulation, deux valeurs de délai ont été utilisées (0ms et 90ms). Les résultats de cette étude sont présentés dans les équations (1.16) et (1.17) ci-dessous.

- Codec *iBLC* :

$$\begin{aligned} QoE_{iLBC}(P_L) &= 2.866 * e^{-26.335 * P_L} + 1.122 \\ QoE_{iLBC}^{0ms}(\rho) &= 2.735 * e^{-12.179 * \rho} + 1.128 \\ QoE_{iLBC}^{90ms}(\rho) &= 2.479 * e^{-10.495 * \rho} + 1.143 \end{aligned} \quad (1.16)$$

- Codec *G.711* :

$$\begin{aligned} QoE_{G.711}(P_L) &= 2.861 * e^{-29.816 * P_L} + 1.134 \\ QoE_{G.711}^{0ms}(\rho) &= 1.084 * e^{-213.298 * \rho} + 1.181 \\ QoE_{G.711}^{90ms}(\rho) &= 0.899 * e^{-149.495 * \rho} + 1.130 \end{aligned} \quad (1.17)$$

Les résultats de cette étude ont montré que le paramètre de réordonnement des paquets reflète mieux la QoE usager de la VoIP que les mesures de la gigue pure.

Une autre étude utilisant l'hypothèse IQX pour le service vidéo à la demande a été réalisée dans [143]. Les tests se sont déroulés dans un environnement contrôlé à l'aide de l'émulateur de réseau NetEm¹¹ (sous Linux). La plateforme utilisée dans les tests consiste en une architecture client-serveur. Le client est un lecteur de flux vidéo VLC¹² et le serveur est un serveur de diffusion vidéo VLC. Les paramètres réseau manipulés sont le délai (*IPTD*) et la perte de paquets (*IPLR*). Pour collecter la base de données subjective, une série d'expériences impliquant 54 individus a été réalisée en utilisant le score MOS pour l'évaluation. Pour calculer la QoE usager, les auteurs ont introduit un nouveau modèle basé sur l'hypothèse IQX afin de quantifier la satisfaction des utilisateurs du service vidéo en termes de valeurs de MOS récoltés auparavant. Les deux paramètres (*IPTD*, *IPLR*) sont utilisés comme présenté dans l'équation suivante (1.18) :

$$QoE_{VoD} = \exp^{(1.038 - (1.081 \cdot 10^{-4}) \cdot IPTD - (4.937 \cdot 10^{-3}) \cdot IPLR)} \quad (1.18)$$

1.6.4 Synthèse

Le Tableau (1.5) ci-dessous présente une synthèse des méthodes qui utilisent la régression statistique pour estimer la QoE usager. La deuxième colonne décrit l'étude réalisée. La troisième indique le modèle mathématique utilisé pour analyser la relation entre la QoE usager et les facteurs qui l'impactent. Les études présentées dans cette synthèse sont classifiées selon ce modèle. La quatrième colonne présente quant à elle les facteurs utilisés comme entrées pour calculer la QoE usager. La dernière colonne décrit quelques commentaires sur chaque étude.

11. <https://www.linux.org/docs/man8/tc-netem.html>

12. <https://github.com/videolan/vlc>

Étude	Description	Modèle	Facteurs	Commentaires
[76]	Utiliser des paramètres applicatifs pour estimer les MOSs usagers.	Linéaire	- Qualité vidéo.	- Les paramètres réseau ne sont pas pris en considération, - Le modèle linéaire peut être amélioré.
[130]	Modéliser la QoE du téléchargement de fichiers en fonction de leur taille normalisée et de la bande passante.	Logarithmique	- Taille fichier - Bande passante réseau	- Bonne performance avec 0.063 d'erreur quadratique. - Taille de fichier doit être connue à l'avance.
[149]	Réaliser une fonction d'ajustement logarithmique pour le taux de résiliation (désabonnement).	Logarithmique	- Bande passante	- Pas de paramètres applicatifs - Les performances de prédiction peuvent être améliorées.
[144]	Étudier les effets de la bande passante et du changement de fréquence de la VoIP.	Logarithmique	- Débit binaire audio	- Validation avec des tests réels de la VoIP. - Modèle spécifique au codec SILK.
[63]	Définir un modèle QoE en termes de score MOS pour différents services multimédia (VoIP, VoD).	Fractionnaire	- Le délai, la gigue, le taux de perte de paquets, les erreurs de paquets et la bande passante.	- Pas de paramètres applicatifs.
[151]	Comparer le modèle puissance avec le modèle logarithmique pour le service vidéo.	Puissance Logarithmique	- Taux de perte paquets	Les fonctions de puissance peuvent évaluer la QoE mieux que les fonctions logarithmiques.
[70]	Proposer un nouveau modèle à base de l'hypothèse IQX pour la VoIP.	Exponentiel	- Gigue - Perte de des paquets - Taux de réordonnement	- Deux codecs (iLBC et G711) testés. - Modèle dépendant du codec et absence de paramètres applicatifs.
[150]	Démontrer que la relation entre les facteurs mesurés et la QoE est sous une forme puissance pour le service streaming vidéo.	Puissance	- Taux de perte des paquets	- Bonne performance (R=0.989 et MSE=0.01) - Besoin de plus de facteurs (application et réseaux)

TABLE 1.5 – Synthèse de méthodes d'estimation de la QoE utilisant la régression statistique

2) Modèles hybrides de la QoE basés sur l'Apprentissage Automatique

L'apprentissage artificiel ou automatique (*Machine Learning*), est une technique qui permet d'utiliser des données existantes afin de prédire des tendances, des résultats et des comporte-

ments futurs [54]. Autrement dit, les algorithmes ont la capacité d'apprendre sans être explicitement programmés, cela grâce aux méthodes d'apprentissage automatique. Parmi ces méthodes figurent les arbres de décision, les réseaux de neurones, les systèmes experts flous, les machines à vecteurs de support (SVM), etc. L'usage de ces méthodes est destiné à toute application qui se base sur l'utilisation des données ou d'observations. En effet, la possibilité de prédire un phénomène à partir d'observations passées ou de données récoltées présuppose l'existence d'un mécanisme causal.

Dans le domaine d'évaluation de la QoE usager des services multimédia, un modèle de calcul de la QoE usager peut utiliser ces méthodes pour trouver l'estimation de la QoE à partir des facteurs mesurés. Le processus d'apprentissage consiste à entraîner le modèle d'estimation avec un ensemble de données subjectives. Une fois que le modèle est appris, il peut prédire la QoE en fonction d'une combinaison de facteurs d'entrée.

Les méthodes d'apprentissage automatique présentent de nombreux avantages[78] :

- (+) Elles assurent plus de précision que les règles créées par l'Homme puisqu'elles sont pilotées par des données.
- (+) Elles ne nécessitent pas d'experts ou de programmeurs pour prendre une décision.
- (+) Elles peuvent fonctionner automatiquement et en temps réel.
- (+) Sont flexibles et peu coûteuses.

Toutefois, si les méthodes d'apprentissage automatique sont efficaces, elles présentent l'inconvénient d'exiger beaucoup d'observations ou données (réelles) pour entraîner le modèle d'estimation, et ces données/observations sont souvent difficiles à récolter sur le terrain.

Dans cette partie, nous présentons quelques études qui ont utilisé les méthodes d'apprentissage automatique pour calculer la QoE usager. Ces méthodes sont les arbres de décisions, les réseaux de neurones et les machines à vecteurs de support (SVM).

*) Arbres de décision

En utilisant les arbres de décision avec une heuristique pour réduire sa taille, Balachandran et al. [16] ont développé un modèle prédictif de mesure de la QoE vidéo sur Internet. Ce modèle vise à prédire le temps d'arrêt de la visualisation de la vidéo sur Netflix¹³. En fait, l'article a présenté une approche axée sur des données subjectives pour modéliser les interdépendances des facteurs et leurs relations complexes avec le contrat *SLA*. En collaboration avec Netflix, une campagne de *crowdsourcing* a été réalisée pour construire la base de données. En utilisant celle-ci, un modèle réduit à base d'arbres de décision a été construit pour anticiper l'instant où les utilisateurs de Netflix vont stopper la vidéo. Les paramètres utilisés dans l'arbre de décision sont divers : le type de connexion, le débit binaire vidéo, le type du terminal et le type du service vidéo (vidéo en ligne ou à la demande). Dans cette étude, une approche intéressante de réduction de la taille de l'arbre de décision a été présentée. Les auteurs expliquent que l'engagement (pour voir une vidéo) est directement ou indirectement influencé par des facteurs de confusion tels que le genre, la popularité, l'appareil, etc. C'est pourquoi une méthodologie a été proposée pour identifier et traiter ces facteurs de confusion. On peut néanmoins déplorer que la base de données n'est pas accessible librement.

Dans [165], Spetebroot et al. ont construit un modèle d'arbre de décision pour calculer la QoE de l'application Skype. Pour exprimer leur ressenti, les participants ont utilisé les valeurs suivantes : "Excellent", "Bon", "Mauvais" et "Aucun avis". Pour émuler les conditions réseau, deux paramètres (débit binaire descendant (téléchargement) et débit binaire ascendant (montant)) ont été variés dans les stimulus. Le modèle d'estimation de la QoE parvient à 83% de taux de classification. Ce travail a pour objectif de permettre aux opérateurs réseau de prévoir la satisfaction de l'utilisateur avant de lancer de nouveaux services, mais la base de données utilisée n'est pas en accès libre aux chercheurs.

En utilisant aussi les arbres de décision, Menkovski et al. ont proposé dans [174], un nouveau modèle pour calculer la QoE en fonction de score MOS. Les auteurs ont testé plusieurs

13. <https://www.netflix.com/fr/>

algorithmes d'arbre de décision différents (ID3, CART, etc.) pour en sélectionner le meilleur. Ils ont trouvé que les plus appropriés dans leurs cas sont les arbres optionnels Hoeffding (Hoeffding Option Trees). Ils ont démontré l'utilité de cette approche en la testant sur les données subjectives qu'ils ont récoltées. Cette étude a montré qu'avec l'utilisation de l'algorithme de Hoeffding, ils ont pu atteindre 90% de taux de classification avec 3500 échantillons. On déplore aussi dans ce cas que la base de données ne soit pas accessible librement.

*) Réseaux de neurones

Du et al. ont proposé dans [62] un modèle qui prédit le niveau de la QoE usager à partir des paramètres QoS du niveau réseau pour le service vidéo. Pour construire celui-ci, les auteurs ont utilisé un réseau de neurones récurrent (*Recurrent Neural Network (RNN)*). Pour commencer, une campagne d'évaluation subjective a été réalisée pour construire une base de données. Celle-ci a été utilisée pour élaborer les paramètres optimaux du RNN utilisé. Plusieurs paramètres réseau (QoS) ont été testés : le délai, la gigue, le taux de perte de paquets, la bande passante, la période de congestion, etc. À la fin de l'étude, les auteurs ont établi un modèle de calcul de la QoE usager basé sur le RNN qui permet d'ajuster les paramètres du réseau (QoS) d'entrée pour obtenir la meilleure satisfaction avec un bon temps d'exécution.

Machado et al. [171] ont présenté une méthode de calcul de la QoE vidéo dans les réseaux WiMAX en termes de scores MOS en fonction des métriques du réseau (QoS). Dans cette étude, la QoE a été calculée en utilisant un réseau de neurones multicouches (ANN), de type *Perceptron multicouches*, construit à partir de données générées au moyen des outils *WEKA* [113], *EvalVid*¹⁴ et *Network Simulator*¹⁵(NS-2). Les facteurs testés dans cette étude sont le délai, la gigue et le taux de perte de paquets. Plusieurs configurations de l'ANN (nombre de couches, nombre de neurones, etc.) ont été testées pour dégager la configuration optimale. À la fin de l'étude, les auteurs ont souligné que la QoE est influencée par le contenu de la vidéo et du nombre de neurones utilisés dans le modèle de calcul de la QoE usager.

14. <http://www2.tkn.tu-berlin.de/research/evalvid/fw.html>

15. <https://www.isi.edu/nsnam/ns/>

En utilisant aussi les ANNs, Calyam et al. [127] ont proposé un modèle en ligne d'estimation de la QoE usager pour le service de télévision sur IP (IPTV). En effet, le modèle est proposé aux fournisseurs de services pour prédire la QoE usager des applications multimédia en temps réel sur la base des mesures d'états du réseau. Plusieurs facteurs ont été testés (débit vidéo, délai, taux de perte de paquets). En outre, les auteurs ont étudié aussi l'impact des techniques de mise en file d'attente au niveau des routeurs (PFIFO (mesuré au niveau des paquets) et TFIFO (mesuré au niveau des octets)). À la fin de l'étude, les auteurs ont établi un modèle prédictif de la QoE ayant plus de 90% de taux de corrélation et moins de 0.7 d'erreur quadratique moyenne par rapport aux scores MOS subjectifs collectés. Concernant la vitesse de traitement, le modèle indique que plus de 10000 flux peuvent être gérés en moins de 4 ms.

En utilisant les réseaux de neurones randomisés (RNNs), l'approche appelée PSQA (*Pseudo-Subjective Quality Assessment*) a été proposée afin de quantifier la qualité d'une séquence vidéo ou audio perçue par l'utilisateur, reçue via un réseau IP. Cette quantification se fait automatiquement, et en temps réel. En utilisant l'approche PSQA, Singh et al. ont proposé dans [83] une étude pour quantifier la QoE usager en termes de MOS pour le service vidéo. Dans cette proposition, aucune référence n'a été utilisée pour construire un module de supervision de la QoE. Ce module est utilisé lors de la transmission adaptative du flux utilisant le codec vidéo H.264 via HTTP. Pour construire le modèle d'estimation, les auteurs ont procédé en deux phases. Dans la première, le RNN apprend la cartographie des configurations internes des facteurs (débit binaire vidéo, taux de perte de compression) et le MOS des usagers tel qu'il est défini dans la phase d'apprentissage. Dans la deuxième phase, le RNN calcule un MOS prédit à partir des paramètres utilisés. Comme résultat, le modèle RNN donne la meilleure prédiction avec une erreur quadratique moyenne de 0,37. Enfin, les auteurs ont expliqué qu'en fonction de la valeur prédite, la vidéo est adaptée dans le module de supervision de la QoE pour qu'elle soit optimisée en fonction des ressources disponibles.

*) **Machines à vecteurs de support**

Hoßfeld et al. [162] ont étudié "*l'effet de mémoire*" comme un facteur d'influence clé sur le calcul de la QoE usager dans le cas d'un service de navigation web. Ce facteur reflète l'influence psychologique de l'historique d'accès à des sites web. Les auteurs de cette étude ont discuté trois approches de modélisation : les machines à vecteurs de support (SVM), la régression exponentielle et le modèle de Markov caché. Ayant utilisé le score MOS dans l'évaluation subjective, les auteurs ont identifié les catégories 1 à 3 de l'échelle MOS comme classe "-1" et la catégorie 4 à 5 en tant que classe "1". Pour tester le modèle SVM, deux tests en ligne sont réalisés en utilisant une validation croisée 10 fois sur les données mesurées. Dans le premier test, le modèle SVM a prédit correctement dans 86.8% des cas le score MOS de l'utilisateur, tandis que dans le second test, le modèle d'estimation SVM a prédit le score MOS correctement dans 82.3%.

Une autre méthodologie qui utilise des tests subjectifs pour développer un modèle d'estimation en utilisant la méthode SVM a été présentée dans [174] pour mesurer la QoE usager du service vidéo. Ce modèle est utilisé dans le cadre d'une plateforme pour la télévision sur mobile. Pour construire le modèle, une campagne de test initial a été réalisée en utilisant des paramètres applicatifs (AQoS) et des paramètres réseau (NQoS). L'AQoS implique des paramètres au niveau de l'application comme le débit binaire vidéo, le débit binaire audio et le taux d'images vidéo. Le NQoS représente les paramètres au niveau réseau comme la perte de paquets, la gigue et le délai de bout en bout. Pour mesurer la QoE des usagers, la construction du modèle s'est déroulée en deux phases. Dans la première phase, un modèle a été construit en utilisant les arbres de décision, pour chaque question (une à neuf), afin de calculer la valeur QoE. Dans la deuxième phase, un modèle utilisant la régression avec SVM, met en corrélation les réponses des questions une à neuf dans une réponse finale pour la question dix (valeur de la QoE). Le modèle final formé sur la base des deux méthodes d'apprentissage automatique utilisées (Arbre de décision et SVM) permet de (i) prédire la perception de l'utilisateur avant de lancer ses applications et (ii) de savoir comment les performances du réseau auront une incidence sur la QoE de l'utilisateur.

*) **Autres modèles d'estimations de la QoE**

En utilisant la logique floue, Alreshoodi et al. [106] ont proposé un modèle pour calculer la QoE usager lors de l'utilisation de différents types de contenus vidéo (caractéristiques spatio-temporelles). Dans cette étude, le score MOS a été utilisé dans la campagne subjective, où une combinaison de paramètres issus du réseau et de l'application a été testée. Dans la partie évaluation, deux méthodes ont été utilisées. La première concerne un système à inférence floue (SIF) et la deuxième un modèle de régression. Les auteurs ont indiqué que le modèle basé sur la logique floue offre une bonne précision de prédiction avec une erreur quadratique moyenne égale à 0.17 comparativement au modèle régressif, ayant une erreur quadratique moyenne égale à 0.37. Par ailleurs, les auteurs ont démontré que les paramètres issus du réseau ont plus d'impact sur la qualité de la vidéo que les paramètres issus de l'application.

Hamam et al. [65] ont proposé une taxonomie pour évaluer la QoE d'un environnement virtuel haptique. Autrement dit, il s'agit de quantifier le ressenti usager (contact physique) lors de la manipulation d'objets virtuels en 3D. Pour modéliser la taxonomie, les auteurs ont proposé un modèle à base de la logique floue en utilisant un système d'inférence floue de type Mamdani avec plusieurs facteurs (le degré de synchronisation, le degré de fatigue et le taux de diminution du degré d'immersion). Plusieurs configurations du système d'inférence floue ont été testées, la plus performante étant celle ayant 10 règles et 5 variables (entrées).

Pokhrel a quant à lui utilisé dans [78] une méthode d'estimation de la QoE usager pour le service vidéo. Pour y parvenir, une campagne de test subjective a été réalisée en testant les facteurs suivants : le taux de perte de paquets, la gigue et les rafales (*bursts* vidéo). Dans la partie évaluation, un système expert a été combiné avec un système d'inférence flou "Sugeno"[187] pour estimer l'impact de ces facteurs sur la qualité du trafic vidéo. Les fonctions d'appartenance au système flou Sugeno sont dérivées des distributions de probabilité normalisées des indicateurs réseau (QoS) corrélées avec la QoE usager. Le modèle proposé a été incorporé dans le

cadre d'un outil de supervision du service IPTV.

En combinant l'utilisation de la logique floue et les RNN, le système adaptatif d'inflexion floue neuronale (en anglais : ANFIS), a été introduit pour estimer la QoE. Ce système est une sorte de réseau neuronal artificiel utilisant un système d'inférence Takagi-Sugeno floue [80]. Selon les auteurs de [20], la procédure d'apprentissage utilisée par le système ANFIS peut également construire une cartographie entrées-sorties (de la relation QoE/facteurs), basée à la fois sur les connaissances humaines (sous la forme de règles flou "Si-Alors") et sur les paires de données entrées-sorties apprises.

Dans l'étude [19], Khan et al ont utilisé un système ANFIS pour évaluer la QoE vidéo sur les réseaux cellulaires. Deux catégories de facteurs ont été utilisées. La première concerne les paramètres applicatifs qui sont le type de contenu (CT), le débit binaire vidéo (SBR) et la fréquence d'image (FR). La deuxième catégorie concerne les paramètres réseau qui sont la largeur de bande de passante (LBW), le taux d'erreur de bloc (BLER) et le taux d'erreur de paquet (PER). En variant tous ces paramètres, une campagne de test subjective a été réalisée pour construire une base de données. Celle-ci est utilisée par le système ANFIS afin d'entraîner le réseau neuronal artificiel utilisant un système d'inférence floue de type Takagi-Sugeno. Dans cette étude, les auteurs ont testé deux codecs (*MPEG4* et *H.264*) et ont déduit que le paramètre applicatif lié au contenu (*CT*) et le paramètre réseau lié à la bande passante (*LBW*) sont les paramètres les plus importants par rapport à leur jeu de données.

Étude comparative

Le Tableau (1.6)) synthétise les études utilisées pour évaluer la QoE en utilisant une approche hybride.

Étude	Campagne de test	Modèle et évaluation	Commentaire
[106]	<ul style="list-style-type: none"> - Méthode : score MOS. - QoE IFs : Taux d'images, débit binaire vidéo, type de contenu vidéo, Taux d'erreur vidéo et durée moyenne de blocage. 	<p>Système inférence floue : RMSE = 0.17 / R = 0.91</p> <p>Modèle régressif : RMSE = 0.37 / R = 0.87</p>	<ul style="list-style-type: none"> - L'aspect temps réel n'a pas été pris en considération. - Le SIF peut être plus performant que la régression dans certains cas.
[62]	<ul style="list-style-type: none"> - Méthode : score DMOS. - QoE IFs : Délai, gigue, taux de perte de paquets, bande passante, période de congestion, temps de blocage et Taux de réordonnement. 	<p>Réseau de neurones multicouches</p> <ul style="list-style-type: none"> - Perceptron avec rétro-propagation 	<ul style="list-style-type: none"> - Utilisation du score DMOS. - Pas de paramètres applicatifs. - Résultats validés avec PSNR et SSIM.
[171]	<ul style="list-style-type: none"> - Méthode : score MOS. - QoE IFs : Délai, gigue et taux de perte de paquets. 	<p>Réseau de neurones :</p> <ul style="list-style-type: none"> - Taux erreur lors de l'apprentissage est 0.10361. - Taux erreur lors du test est 0.1414 	<ul style="list-style-type: none"> - Pas de paramètres applicatifs. - La QoE est influencée par la dynamique vidéo et du nombre de neurones utilisés dans le modèle prédictif.
[165]	<ul style="list-style-type: none"> - Méthode : QoE présente le temps d'arrêt de visionnage d'une vidéo. - QoE IFs : Temps de téléchargement, temps d'upload, délai, la bande passante et taux de perte de paquets. 	<ul style="list-style-type: none"> - Arbre de décisions. - 83% de taux de classification 	<ul style="list-style-type: none"> - Combinaison de paramètres applicatifs et réseau. - La QoE est présentée sur 4 niveaux : (Excellent, Bon, Mauvais et Aucun avis).
[78]	<ul style="list-style-type: none"> - Méthode : score MOS. - QoE IFs : Taux de perte paquets, gigue et durée moyenne de blocage. 	<ul style="list-style-type: none"> - Système inférence floue Sugeno. - Dans 60% des tests, les différences de score étaient < 0, 5. - Dans 83% des tests, les différences de score étaient < 1. 	<ul style="list-style-type: none"> - Durée de l'expérience énorme (environ 2, 5 heures). - Peut ajouter de la bande passante au QoE IFs.
[162]	<ul style="list-style-type: none"> - Méthode : Score MOS mais sur deux classes (satisfait (4,5), insatisfait (1,2,3)). - QoE IFs : Temps de chargement d'une page web et l'effet de l'historique de la navigation web. - Support Vector Machine (SVM). 	<ul style="list-style-type: none"> - Test sur deux configurations. - Test 1 : 86, 8% de taux de classification. - Test 2 : 82, 3% de taux de classification. 	<p>Le SVM proposé ne peut pas calculer un score MOS réel (5 niveaux) car il traite la QoE tant qu'un problème de deux classes (insatisfait et satisfait).</p>
[16]	<ul style="list-style-type: none"> - Méthode : QoE présente le temps d'arrêt de visionnage d'une vidéo. - QoE IFs : Type de connexion, débit binaire vidéo, type du terminal et le type du service vidéo (vidéo à la demande ou vidéo en ligne). 	<ul style="list-style-type: none"> - Arbre de décisions (taux de classification) : - Arbre de décision simple = 45, 02% - Arbre de décision multiple = 68, 74% 	<ul style="list-style-type: none"> - Méthode intéressante de réduction des arbres. - Le taux de classification reste faible.

Étude	Campagne de test	Modèle et évaluation	Commentaire
[174]	<ul style="list-style-type: none"> - Méthode : Deux classes (satisfait, insatisfait). - QoE IFs : débit binaire vidéo, débit binaire audio, taux d'images vidéo, type du terminal. 	<ul style="list-style-type: none"> - Arbre de décisions (taux de classification) - Avec 100 exemples (apprentissage) = 80%. - Avec 3500 exemples (apprentissage) = 90%. 	<ul style="list-style-type: none"> - Cette approche confirme la nécessité d'un grand nombre de tests subjectifs. - Pas de paramètres réseau.
[43]	<ul style="list-style-type: none"> - Méthode : score MOS. - QoE IFs : Type du réseau (GSM ou UMTS), Codec, Direction de la voix. 	<ul style="list-style-type: none"> - Système d'inflexion floue neuronale adaptatif (ANFIS) - Plusieurs SIF testés 	<ul style="list-style-type: none"> - Confirme que la relation QoE/QoE IFs est non-linéarité (mobiles). - Le modèle est testé dans LTE (4G).
[19]	<ul style="list-style-type: none"> -Méthode : score MOS. - QoE IFs : "Réseau" : Bande passante et taux de perte de paquets "Application" : Taux images vidéo et débit vidéo (débit binaire) 	<ul style="list-style-type: none"> - ANFIS : taux de corrélation est de 0.87. - Modèle Régressif : taux de corrélation est de 0.86. 	<ul style="list-style-type: none"> - Utilisation des paramètres applicatifs et réseau. - Dans certains cas, le système ANFIS donne de meilleurs résultats que le modèle régressif.
[20]	<ul style="list-style-type: none"> -Méthode : score MOS. - QoE IFs : "Réseau" : Bande passante, taux de perte de paquets/taux de perte de bloc de données. "Application" : Type de Contenu, taux images vidéo et débit vidéo (débit binaire). 	<ul style="list-style-type: none"> -Système d'inflexion floue neuronale adaptatif (ANFIS) : - Taux de corrélation est : 0.90 	<ul style="list-style-type: none"> Le type de contenu (application) et la bande passante (réseau), sont les facteurs les plus impactant.
[143]	<ul style="list-style-type: none"> -Méthode : score MOS. - QoE IFs : Taux de perte paquets et latence (délai). 	<ul style="list-style-type: none"> Hypothèse IQX : - Équation (1.18) 	<ul style="list-style-type: none"> - Introduit pour la première fois l'Hypothèse IQX pour la vidéo. -Nombre de paramètres réduit et absence de paramètres applicatifs.
[83]	<ul style="list-style-type: none"> -Méthode : score MOS. -QoE IFs : Débit vidéo (débit binaire), taux de perte de compression, Résolution. 	<ul style="list-style-type: none"> - Réseau de neurones multi-couches (3 couches et 5 neurones dans la couche cachée). - Erreur quadratique (RMSE) = 0,37 	<ul style="list-style-type: none"> - Les paramètres réseau ne sont pas pris en considération. - La QoE vidéo peut être impacté par le taux de perte de compression.
[127]	<ul style="list-style-type: none"> -Méthode : score MOS. -QoE IFs : Vidéo débit binaire, délai, taux de perte, type de la file au niveau routeur 	<ul style="list-style-type: none"> Réseau de neurones multicouches avec un taux de corrélation (R) de 0.9 et un RMSE < 0.7 	<ul style="list-style-type: none"> - Combinaison des paramètres applicatifs et réseau. - Bonne vitesse de traitement (plus de 10000 flux chaque 4 ms).

TABLE 1.6 – Synthèse d'études d'évaluation de la QoE usager en utilisant une approche hybride

Dans cette synthèse (Tableau (1.6)), la description de l'évaluation subjective réalisée est présentée en premier lieu dans la "Colonne 2". Cette évaluation se base sur la collecte d'un nombre de facteurs (notés : "QoEIFs") et sur l'utilisation d'un score subjectif. En deuxième lieu, l'évaluation objective de la QoE est illustrée dans la "Colonne 3". Cette évaluation repose sur l'utilisation de méthodes statistiques ou des méthodes issues de l'apprentissage automatique. Les méthodes statistiques basées sur un seul paramètre sont faciles à mettre en place, mais ne permettent pas de prendre en compte un grand nombre de facteurs. Les méthodes statistiques multi-paramètres sont intéressantes, mais sont actuellement trop orientées sur des paramètres du niveau réseau (QoS). À cet égard, l'utilisation des méthodes d'apprentissage automatique devrait être encouragée [26, 31]. En effet, ces méthodes sont intéressantes car elles se reposent sur des jeux de données réels collectés auprès d'utilisateurs et elle peuvent prendre en compte un grand nombre de facteurs. Cependant, elles peuvent être éloignées des besoins personnels si les paramètres de simulation utilisés ne reflètent pas le contexte d'utilisation de l'application dédiée. Enfin, quelques commentaires sont formulés pour discuter chaque étude.

1.7 Conclusion

Comprendre et estimer la valeur de la QoE usager est devenue essentiel pour les fournisseurs de services qui souhaitent fidéliser leurs utilisateurs. En effet, c'est sur elle que repose la mise en œuvre de nouvelles politiques de supervision, dites "orientée QoE", complémentaire de la supervision orientée réseau. Dans ce chapitre, nous avons introduit, dans la première partie, le concept de QoE et proposer une nouvelle classification de facteurs en cinq catégories (les facteurs applicatifs, les facteurs réseau, les facteurs du terminal, les facteurs du profil usager et les retours usager). Dans la seconde partie de ce chapitre, nous avons présenté les méthodes, subjectives et objectives, de mesure de la QoE. Enfin, nous avons établi un état de l'art ainsi qu'une étude comparative des travaux d'évaluation de la QoE usager. Dans le chapitre suivant, nous présenterons nos campagnes de tests subjectifs réalisées afin de collecter des données réelles. Ces données seront par la suite utilisées dans la construction de notre modèle d'estimation de la QoE usager.

Chapitre 2

Campagnes subjectives pour l'évaluation de la Qualité d'Expérience usager

2.1 Introduction

Les campagnes de test subjectives sont importantes pour évaluer la QoE usager. En effet, elles permettent de collecter des échantillons de données réelles, qui seront utilisées par la suite pour construire des modèles de calcul de la QoE usager. Cependant, la construction de telles bases de données est une tâche difficile, vu la nécessité de mobiliser des ressources humaines et matériels, en plus du temps et du coût financier. De plus, une fois construite, ces données ne sont pas largement diffusées. Toutes ces raisons font que nous avons dû construire nos propres bases de données.

L'objectif de ce chapitre est de décrire les campagnes de test que nous avons réalisées pour construire nos propres bases de données, qui nous serviront à évaluer la QoE usager d'un service vidéo. Pour cela, nous commençons par présenter un certain nombre d'études qui ont élaboré des tests subjectifs dans un environnement contrôlé et non-contrôlé. Ensuite, nous détaillons la campagne de test, en environnement contrôlé (laboratoire). Enfin, nous présentons la campagne

de test non-contrôlé (*crowdsourcing*).

2.2 Environnements de l'évaluation subjective de la QoE

Il existe deux catégories d'approches pour évaluer la QoE d'un service multimédia. La première est l'approche subjective. Elle est utilisée pour évaluer la qualité directement par les humains. Cette évaluation peut soit être faite dans des environnements contrôlés (laboratoires, maisons, etc.) ou dans des environnements non-contrôlés (*crowdsourcing*). La deuxième consiste dans l'évaluation objective de la QoE sans intervention humaine. La Figure (2.1) présente ces deux catégories d'approches d'évaluation de la QoE.

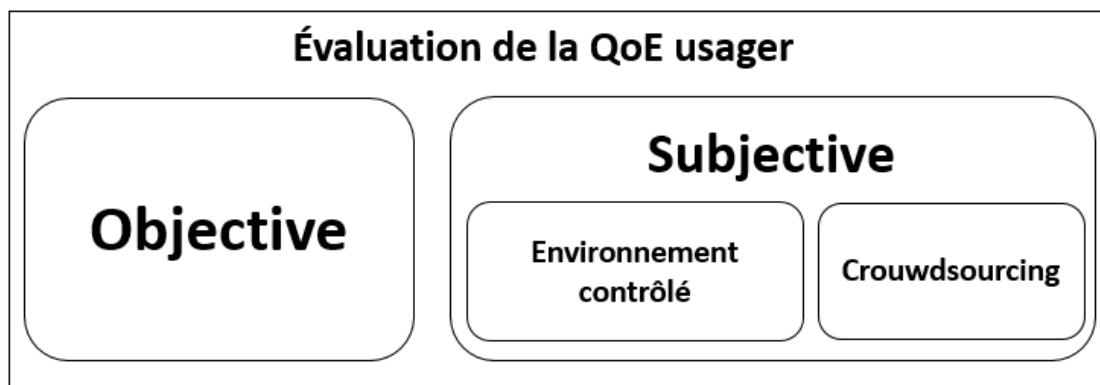


FIGURE 2.1 – Méthodes d'évaluation de la QoE

Ce chapitre se focalise sur l'évaluation subjective de la QoE. Il présente nos deux campagnes de test subjectives. La première concerne la campagne de test dans l'environnement contrôlé au sein de notre laboratoire LiSSi¹. La deuxième est réalisée dans un environnement non-contrôlé dans différents emplacements en utilisant le réseau Internet. Le but de ces campagnes de test est d'avoir des observations qui nous permettront ensuite de construire des échantillons de données. Ces derniers seront utilisés pour construire de meilleurs modèles d'estimation de la QoE usager.

1. Laboratoire Images, Signaux et Systèmes Intelligents (<https://lab.lissi.fr/>)

2.2.1 Environnement contrôlé

D'après la norme ITU-T Rec. E.800 [8], la qualité de l'expérience exprime le ressenti final à travers les mesures prises au niveau des utilisateurs. Par conséquent, la méthode subjective d'évaluation de la QoE consiste à faire participer des humains dans des expériences contrôlées et/ou non-contrôlées pour mesurer la QoE usager. Ces méthodes sont souvent appelées évaluations subjectives de la qualité.

La première approche que nous aborderons dans cette partie concerne l'évaluation subjective de la QoE dans un environnement contrôlé. Ce type d'évaluation est généralement utilisé dans un emplacement spécialement conçu (laboratoire, maison, etc.) pour fixer et varier les facteurs environnementaux qui peuvent influencer la QoE usager. L'ITU-T a défini les recommandations visant à mettre en place ce type de campagne d'évaluation. Par exemple, la recommandation [7] fournit les lignes directrices pour la conduite d'une campagne de test subjective dans un environnement contrôlé, y compris la sélection des participants qui représentent les utilisateurs d'un service. En effet, pour obtenir la notation subjective (score subjectif comme le *MOS*[137]) conformément à la recommandation ITU-T [4], quelques participants devraient être des non-experts. Autrement dit, ils ne devraient pas être directement concernés par la qualité de l'image ou de la vidéo dans le cadre de leur travail quotidien. Les avantages de cette approche sont :

- L'environnement de test est totalement sous contrôle.
- L'analyse aisée de l'influence d'un paramètre individuel.
- La liberté de choisir les participants qui appartiennent à des milieux, professions, groupes d'âge, etc. différents.

Les utilisateurs humains sont les destinataires finaux de la plupart des applications proposées (vidéo streaming, voix sur IP, téléchargement, etc.). À cet égard, l'évaluation subjective de l'expérience utilisateur est le moyen le plus précis et le plus fiable. Cependant, cette évaluation subjective de la QoE dans un environnement contrôlé comporte également certaines limites qui sont résumées ci-dessous :

- Ils sont des tests qui consomment beaucoup de temps. Par conséquent, ils ne sont pas exécutables en temps réel.

- Il est difficile de mobiliser un nombre important de participants disposés à passer du temps en laboratoire et à exprimer leur perception de la qualité du service vidéo.
- Il s'agit d'une approche coûteuse dans la mesure où il faut acheter les équipements nécessaires à l'exécution du test.
- Il est difficile de configurer l'environnement de manière à émuler le monde réel.

2.2.2 Environnement non-contrôlé

La deuxième approche d'évaluation subjective est l'évaluation dans un environnement non contrôlé, connu aussi sous l'appellation *crowdsourcing*. Dans cette approche, une tâche d'évaluation d'un service multimédia est attribuée à un grand nombre d'utilisateurs anonymes qui peuvent y participer via Internet. Par conséquent, le crowdsourcing est une approche efficace dans laquelle les ensembles de données recueillis représentent l'opinion d'un grand nombre d'utilisateurs d'un service donné. Pour ce qui est de ses avantages, on peut citer :

- Un environnement représentatif et diversifié.
- La collecte aisée d'un grand échantillon de données.
- La participation d'un grand nombre de participants.
- Le recueil des paramètres en temps réel.
- La réduction du coût de mise en œuvre de la campagne de test.

Néanmoins, l'approche utilisant le crowdsourcing a également quelques inconvénients que nous résumons ci-dessous.

- Un environnement différent pour chaque participant.
- Les résultats sont plus influencés par les conditions physiques et émotionnelles des testeurs.
- Nécessite l'installation du logiciel sur l'appareil de l'utilisateur final.
- Nécessite une description ou une formation pour chaque participant afin d'exécuter l'évaluation concrètement.

2.2.3 Base de données existantes

Pour évaluer la QoE usager, deux écoles existent. Celle qui plaide pour une évaluation subjective en demandant explicitement à l'utilisateur son ressenti et celle qui plaide pour une évaluation objective et non intrusive en utilisant les outils de l'apprentissage automatique. Cette section s'inscrit dans le premier cas de figure. Nous présentons dans le Tableau (2.1) une synthèse d'études qui ont élaboré des tests subjectifs dans un environnement contrôlé (*EC*) ou non-contrôlé (*ENC*). D'autres études ont élaboré aussi de larges bases de données évaluant la QoE, mais elles ne sont pas encore en accès libre telles que [16] et [34].

La synthèse que nous avons réalisé nous a permis d'identifier trois éléments clés : la méthode subjective décrivant le processus d'évaluation, les testeurs qui vont participer à la campagne d'évaluation et l'environnement de test. Bien qu'il ne soit pas possible ici d'évaluer intégralement les trois éléments, un certain nombre de remarques peuvent cependant être formulées.

À partir de cette synthèse, on remarque que la majorité des études ont utilisé la méthode subjective "ACR" (Section 1.5.2), car elle est simple, pratique et facile à mettre en place[2]. En outre, on trouve aussi que la note d'évaluation la plus utilisée est le score MOS[137] avec 5 niveaux de qualité (excellente, bonne, moyenne, mauvaise et très mauvaise).

Concernant le nombre d'utilisateurs, le nombre de sujets recommandés par la P800 de l'ITU-T [2] est de 4 à 40, avec certains testeurs non-experts. D'après le groupe de travail VQEG², le nombre de testeurs mobilisés pour les tests doit être supérieur à 24. À partir du Tableau (2.1), on observe qu'un minimum de 10 testeurs a été utilisé dans l'étude [157]. Par ailleurs, on peut remarquer aussi que le nombre de testeurs est souvent proportionnel au nombre utilisé de configurations de test, comme dans le cas de l'étude [38] avec 100 personnes mobilisés pour évaluer 114 configurations de test (57 chaînes télévisées et 2 types de sélection de chaînes).

Enfin, on peut tirer comme renseignement du Tableau (2.1), le fait que la collecte des données issues de campagnes de test subjectives est une tâche complexe. Ceci est dû au fait qu'il est difficile de solliciter un grand nombre de testeurs. Par conséquent, il est important que les chercheurs s'orientent vers la publication de leurs données afin de favoriser la réutilisation et la reproductibilité, ce qui est notre objectif dans ce chapitre.

2. <https://www.its.bldrdoc.gov/vqeg/vqeg-home.aspx>

Étude	Type	Description	Méthode	Sujets	Campagne de test
[123]	EC	Analyser la QoE des usagers des applications multi-médias immersives, y compris le contenu vidéo UHD.	ACR (9 notes)	20	- Trois types de qualités (SD,HD,UHD), - 27 conditions, - 76 tests.
[126]	EC	Proposer un modèle de prédiction qui estime, en temps réel, la QoE usagers du service vidéo dans les réseaux VANETs.	ACR (5 notes)	/	- Cinq paramètres techniques (réseau et vidéo), - Trois paramètres non-techniques, - 230 exemples récoltés.
[157]	EC	Étudier l'impact d'une optimisation du trafic réseau sur la QoE des joueurs en ligne.	CCR (7 notes)	10	- Sept conditions de test, - Cinq valeurs de débit, - Quatre étapes (périodes) pour les tests.
[140]	ENC	Introduire une mesure subjective de qualité (QoE) appelée "crowdMOS" pour évaluer la qualité de l'image.	ACR (5 notes)	44	- Tests réalisés par des collaborateurs (workers), - 12 images évaluées dans une session de test, - Différents types de contenus sont testés.
[18]	EC	Proposer un nouveau modèle de prédiction de la QoE vidéo incluant plusieurs types de contenu.	ACR (5 notes)	20	- Trois types de contenu vidéo, - 315 séquences vidéo, - Utilisation des paramètres application et réseau.
[105]	EC	Introduire une méthode qui utilise les techniques d'apprentissage artificiel pour construire un modèle de prédiction QoE du service vidéo.	ACR (Oui/Non)	/	- Trois terminaux de test, - Tester les méthodes arbre de décision et SVM, - Utiliser les paramètres réseau et vidéo.
[51]	EC	Élaborer un modèle de prédiction QoE (vidéo) sans référence (NF) en utilisant des systèmes d'inférence floue (SIF).	ACR (5 notes)	21	- 63 conditions de test pour les vidéo HD, - Trois types de contenu vidéo, - Test le codec HEVC (high-efficiency-video-coding).
[143]	EC	Proposer un modèle de corrélation globale entre la QoE et la QoS concernant le service de vidéo à la demande (VoD).	ACR (5 notes)	/	- 900 conditions de test, - Deux paramètres QoS (Délai et taux de perte), - Une seule séquence vidéo.
[45]	ENC	Proposer des modèles d'estimation de la qualité d'expérience en utilisant les méthodes d'apprentissage artificiel dans les réseaux cellulaires.	ACR (Oui/Non).	/	- 700 exemples collectés, - Différents types de fournisseurs de service, - Utiliser 13 paramètres réseau et application ainsi que 14 KPI.
[38]	EC	Étudier l'impact du changement de chaîne (zapping) sur la QoE des usagers de la TV.	ACR (5 notes)	100	- 57 chaînes de télévision, - Deux types de sélection des chaînes TV, - Élaboration d'une étude statistique.
[168]	EC	Comparer les résultats expérimentaux subjectifs obtenus à partir d'une méthode ACR normalisée et semi-continue.	ACR (5 notes)	66	- Évaluation des longues séquences vidéo, - Sept séquences vidéo, - Quatre conditions de codage vidéo.

TABLE 2.1 – Bases de données subjective de la QoE

2.3 Campagne d'évaluation de la QoE dans l'environnement contrôlé

Typiquement, une évaluation subjective de la QoE dans un environnement contrôlé consiste en un ensemble de testeurs humains auxquels on attribue une échelle de notation pour un service et à qui on demande de classer la qualité selon cette échelle. L'évaluation donnée par les testeurs est basée sur le ressenti personnel de chacun concernant la qualité du service multimédia présenté [137]. Cependant, il n'est pas facile d'établir une telle évaluation subjective dans la mesure où il faut (i) mettre en place une plateforme pour interroger les testeurs et (ii) de collecter un ensemble de données (contenant les facteurs et les scores MOS) que nous utiliserons par la suite pour construire le modèle de calcul de la QoE. Dans ce qui suit, nous présenterons la méthodologie d'évaluation subjective que nous avons utilisée pour évaluer la QoE des usagers de la vidéo.

2.3.1 Mise en œuvre de la plateforme de test

Pour l'évaluation de la QoE en environnement contrôlé, nous avons développé une plateforme open-source [94] basée sur un lecteur vidéo VLC³ et le langage de programmation Python⁴. Notre objectif est de collecter un ensemble de données subjectives qui rassemble plusieurs facteurs et de calculer ensuite la QoE usager en utilisant ces facteurs. Dans le cadre de ce chapitre, les facteurs choisis concernent les paramètres de la vidéo (catégorie "QoA" de la classification représentée par la Figure (1.2)) et le score subjectif choisi est le score MOS[137]. Celui-ci consiste en une valeur entière de 1 à 5, où 5 indique une excellente qualité, et 1 représente la pire. La plateforme utilise différents outils et logiciels dédiés pour le contrôle de l'environnement de test. La Figure (2.2) présente l'ensemble des composants de la plateforme et les logiciels utilisés [92].

Cette plateforme consiste en une application développée et installée sur un dispositif utilisateur. Elle est composée des éléments suivants :

3. <https://github.com/videolan/vlc>

4. <https://docs.python.org/3/>

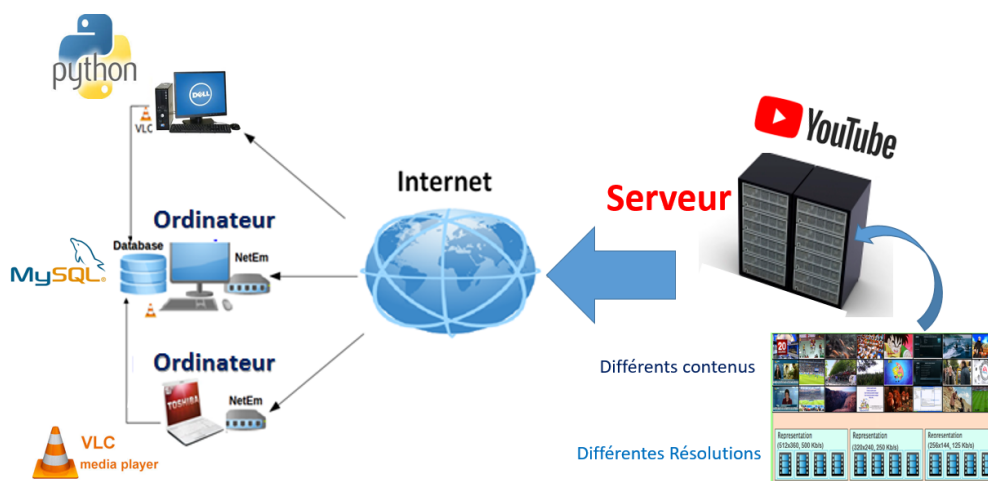


FIGURE 2.2 – Campagne de test réalisée dans un environnement contrôlé

- VideoLAN Client Player (VLC⁵) : le lecteur vidéo utilisé pour afficher la vidéo YouTube en utilisant son URL (Uniform Resource Locator).
- Netem⁶ : l’émulateur réseau qui permet de générer les variations des paramètres de QoS (bande passante de liaison, taux de perte de paquets, etc) dans l’environnement de test.
- Module Python⁷ : le langage de programmation utilisé pour implémenter l’interface graphique qui permet d’interagir avec le testeur et d’envoyer les données collectées (facteurs + MOS) dans la base de données subjective.
- MySQL database server⁸ : pour stocker les valeurs des facteurs mesurés ainsi que les scores MOS des testeurs.
- Plusieurs résolutions vidéo sont considérées allant de 256x144 au Full HD (1920x1080) et divers types de vidéos sont utilisés (sport, film, documentaire, musique, etc.).

Durant les expériences réalisées, chaque testeur utilise la plateforme développée [94] pour exprimer son ressenti personnel après avoir visionné un ensemble de vidéos aléatoires. Chaque participant doit suivre trois étapes : Au début, l’utilisateur répond à quelques questions (une seule fois) afin de construire son profil (facteurs QoU (Figure 1.2)). Ensuite, l’utilisateur peut visionner

5. <https://github.com/videolan/vlc>

6. <https://wiki.linuxfoundation.org/networking/netem>

7. <https://docs.python.org/3/>

8. <https://www.mysql.com/>

une séquence vidéo. Lors de cette étape, le testeur est assis sur une chaise à une distance de 50 centimètres au maximum de l'appareil (ordinateur, téléviseur ou ordinateur portable). Enfin, chaque participant répond à d'autres questions à la fin de chaque séquence vidéo afin de recueillir son retour (facteurs de la catégorie "QoF" de la Figure 1.2). La démo publiée sur YouTube [28] donne un aperçu d'une session en cours d'exécution.

2.3.2 Collecte de la base de données

Pour mener à bien notre d'évaluation subjective, nous nous sommes inspirés des recommandations de l'ITU-T et le groupe QUALINET, qui ont défini un nombre de conditions à respecter dans [4, 102]. Ces recommandations peuvent être résumées en trois phases : (i) Sélection d'un nombre minimum de spectateurs (sujets), (ii) diffusion des séquences vidéo dans des conditions contrôlées et (iii) évaluation de la séquence vidéo par chaque testeur. En se basant sur ces recommandations, notre campagne d'évaluation de la QoE est réalisée en tenant compte des aspects résumés ci-dessous.

- **Conditions d'observation** : Les conditions d'observations dépendent du contexte d'utilisation. Ces conditions sont identiques pour tous les testeurs dans notre campagne de test. Comme exemple, on cite : la position assise, la distance par rapport au dispositif d'affichage, l'utilisation des mêmes équipements de test (ordinateur de bureau ou ordinateur portable, etc).
- **Réplication de séquences vidéo** : Les mêmes séquences vidéo avec les mêmes paramètres ont été montrées aux sujets. L'ordre d'apparition des séquences vidéo a été aléatoire.
- **Participants** : Le nombre de sujets recommandés par la P800 de l'ITU-T [2] est de 4 à 40. Notre panel de test est composé de 62 personnes.
- **Questions** : Les questions utilisées sont simples comme le suggère le rapport QUALINET[102]. Par exemple : Notez la qualité de la vidéo d'une façon globale.
- **Matériel de test** : Un ensemble de vidéos a été utilisé en vue d'étudier l'influence du contenu sur la QoE usager. Ces vidéos sont encodées en utilisant AVC/H.264⁹. Chaque

9. <https://www.divx.com/en/software/technologies/h264>

vidéo a un degré de mouvement spécifique et des résolutions différentes (allant de 256x144 au Full HD (1920x1080)). La Figure 2.7 présente des captures d'écran des vidéos utilisées et l'Annexe (1) présente les caractéristiques détaillées de ces vidéos.

Type de contenus

	Journal	Sport	Documentaire	Musique	Dessins animés	Cours	Film	Jeux
Degré de mouvement Peu								
Moyen								
Grand								

FIGURE 2.3 – Captures d'écran des vidéos utilisées

- **Durée de test** : Nous avons considéré une durée de 30 secondes pour chaque séquence vidéo et nous avons donné aussi aux testeurs la possibilité de mettre fin à une session de test quand ils le souhaitent (à la fin de chaque séquence vidéo).

Cette campagne a permis de construire une base de données contenant 560 exemples. Dans cette base, nous avons recueilli 19 facteurs. Chaque facteur appartient à une catégorie (QoA, QoS, QoD, QoF et MOS) selon notre classification. Les participants aux tests sont des étudiants et des chercheurs du laboratoire *LiSSi*¹⁰. Leur âge est compris entre 19 et 36 ans. Certains ont peu ou pas d'expérience dans l'évaluation de la vidéo. 15% des participants aux tests ont vu au moins dix vidéos durant une session de test et 23% de tous les participants étaient des femmes.

2.4 Campagne d'évaluation de la QoE dans l'environnement non-contrôlé

Dans cette section, nous présentons la deuxième approche d'évaluation de la QoE appelée "crowdsourcing", consistant à évaluer la QoE dans un environnement non-contrôlé.

10. Laboratoire Images, Signaux et Systèmes Intelligents (<https://lab.lissi.fr/>)

2.4.1 Projet PoQEMoN

La campagne de crowdsourcing que nous avons réalisée dans le cadre de cette thèse rentre dans le cadre du projet PoQEMoN¹¹ (*Platform for Quality Evaluation of Mobile Networks*). Ce projet labellisé par le pôle de compétitivité System@tic, a réuni au sein du consortium les partenaires suivants : IP Label, leader du consortium, Orange, ALTRAN, Data-Publica, INRIA, Genymobile, le Groupe Next Radio, HEC et notre équipe de recherche TINCNET¹² (*Team of Intelligent Control for Networks*).

Le projet PoQEMoN vise à étudier les corrélations qui peuvent exister entre le ressenti de l'utilisateur et le service fourni dans les environnements mobiles (3G(UMTS), HSPDA, 4G(LTE-A), etc.). Ces enjeux sont multiples et peuvent être synthétisés comme suit :

- Fournir aux opérateurs réseaux un outil pour mieux appréhender la qualité de service qu'ils délivrent à l'utilisateur. Pour y parvenir, des outils fiables et économiques sont développés dans le cadre de campagnes d'audit (couverture, analyse de problèmes, benchmark).
- Aider les entreprises à améliorer la performance des accès aux systèmes d'information dans un contexte de mobilité (choix du bon opérateur, amélioration de l'infrastructure en certains points de concert avec l'opérateur, etc.).
- Permettre aux consommateurs de comprendre la qualité de service réellement délivrée par les opérateurs mobiles, en fournissant des indicateurs de qualité compréhensibles dans le but de les aider à faire un choix basé non seulement sur le prix de la prestation, mais aussi sur la qualité en fonction de leurs besoins et usages.

2.4.2 Application de test Android

Pour mener à bien notre campagne de test crowdsourcing, une application a été conçue et développée au sein du consortium. Celle-ci se base sur le système d'exploitation mobile Android¹³ et elle vise à récolter les facteurs et les retours des usagers en termes de scores MOS. Dans cette

11. <https://www.ip-label.fr/emailing/newsletter/2013/41/FR/expert.html>

12. <https://tincnet.fr/>

13. <https://www.android.com/>

application, le lecteur VLC a été intégré pour mesurer les valeurs des facteurs du service vidéo (comme le débit binaire vidéo et le nombre d'images vidéo par seconde). L'interface présentée dans la Figure (2.4) présente les questions utilisées.

The figure displays two screenshots of a mobile questionnaire application. Both screenshots show a status bar at the top with icons for connectivity and battery, and the time 14:59. The left screenshot is titled "Questionnaire" and "Q1 - Cerner le profil utilisateur". It contains the following questions: Q1.1 - Age: 42; Q1.2 - Sexe: Homme (selected) / Femme; Q1.3 - Niveau d'étude: Supérieur; Q1.4 - Connaissance des plateformes de vidéos en ligne (est-ce que vous visionnez fréquemment des vidéos sur Internet?): oui (selected) / non; and Q2.5 - Qualité de la voix: Bonne. The right screenshot is also titled "Questionnaire" and "Q2 - Avis utilisateur après visionnage vidéo". It contains the following questions: Q2.1 - Rapidité de lancement de la vidéo: Acceptable; Q2.2 - Avez-vous ressenti de la gêne occasionnée par un décalage entre l'audio et la vidéo?: Aucune gêne; Q2.3 - Qualité de la vidéo (de façon globale): Moyenne; Q2.4 - Qualité de l'image: Bonne; and Q2.5 - Qualité de la voix: Bonne. Both screenshots have an "Envoyer" button at the bottom.

FIGURE 2.4 – Questions utilisées lors de la campagne de crowdsourcing

Dans cette interface, la première partie des questions est présentée une seule fois au début pour cerner le profil du participant au test. Ces questions appartiennent à la catégorie "Profil usager" notée "QoU" (Section 1.4). Elles respectent les recommandations de groupe QUALINET[102] où elles sont courtes et simples.

La deuxième partie des questions concerne l'avis des utilisateurs sur la séquence vidéo visualisée. Dans cette partie, cinq questions ont été définies dont celle qui nous donne le score MOS (qualité de la vidéo globale). En outre, les questions de cette deuxième partie sont affichées à la fin de chaque séquence vidéo comme suggérée par la méthode subjective ACR[2] que nous avons utilisée dans cette campagne de test crowdsourcing.

2.4.3 Campagne de test crowdsourcing

La Figure (2.5) présente un aperçu général de notre campagne de test dans laquelle l'application Android présentée auparavant a été utilisée.

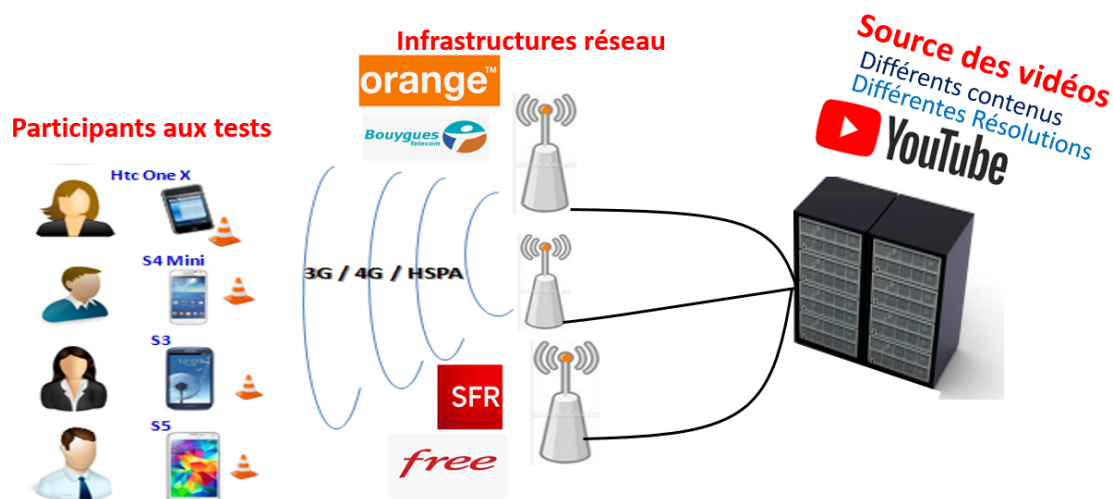


FIGURE 2.5 – Campagne de test crowdsourcing

La campagne de test se compose principalement des éléments ci-dessous :

- Développement d'un lecteur de flux vidéo (VLC) qui permet d'afficher la séquence vidéo et de collecter les informations relatives à la qualité du réseau au niveau du terminal en utilisant le système d'exploitation Android.
- Utilisation de plusieurs types de dispositifs d'affichage de type tablette ou smartphone. Ces dispositifs sont représentatifs des terminaux utilisés par le grand public dans les réseaux cellulaires, à cause de la diversité de leurs caractéristiques techniques telles que la mémoire, la résolution d'écran, la taille de la fenêtre d'affichage, etc.
- Formation des utilisateurs à la procédure de tests et à la méthodologie d'évaluation (comme l'annexe 1).
- Préparation d'un questionnaire à remplir par l'utilisateur, qui permet de cerner son profil et de récolter l'évaluation de la qualité de visionnage.
- Utilisation de plusieurs catégories de contenus vidéo (sport, film, musique, etc.).

Dans cette campagne d'évaluation, différents emplacements urbains et moyens de transport ont été testés pour évaluer le service de diffusion vidéo YouTube. Chacun de ces emplacements est caractérisé par différents facteurs de la QoE comme la puissance du signal de réception (*Received Signal Strength (RSS)*) et le type du réseau utilisé (3G, HSPA, 4G, etc.). Quelques endroits considérés lors de la campagne de crowdsourcing sont présentés dans la Figure (2.6) ci-dessous.



FIGURE 2.6 – Emplacements testés lors de la campagne de crowdsourcing

Concernant le déroulement des tests, la même méthodologie utilisée à la campagne d'évaluation dans le laboratoire (environnement contrôlé) a été reconduite. En effet, les tests se déroulent sous la forme de sessions où un ensemble de vidéos, ayant une durée de 60 secondes, sont à visionner. Lorsqu'une vidéo se termine, le testeur fournit un résultat en termes de score MOS, qui est stocké via un service web dans une base distante de données. Ainsi et en se basant sur les recommandations de l'ITU-T et le groupe de recherche QUALINET, notre campagne d'évaluation crowdsourcing de la QoE usager est réalisée en tenant compte de plusieurs aspects résumés ci-dessous.

- **Conditions d'observation** : Les conditions dépendent de l'emplacement où le test est réalisé. Ces conditions sont variables pour tous les participants car les tests sont réalisés dans différents emplacements comme les stations/gares des moyens de transport, les

- rues, etc. Par ailleurs, quelques explications du déroulement du test ont été données aux participants en amont. Comme exemple on peut citer la distance de l'écran lors du test, qui doit être moins de 3 fois la taille de l'écran d'affichage.
- **Séquences vidéo** : Les mêmes séquences vidéo ont été utilisées par tous les testeurs. L'ordre d'apparition des séquences vidéo été aléatoire afin de ne pas impacter les scores donnés par les testeurs.
 - **Participants** : Notre panel de test est composé de 181 participants. Cela représente un échantillon plus important que la recommandation P800 de l'ITU-T [2] qui préconise un panel entre 4 à 40 testeurs et la recommandation de VQEG qui suggère l'utilisation d'au moins 24 testeurs.
 - **Questions** : Les questions utilisées sont présentées dans la Figure (2.4). Ces questions sont simples à comprendre et constituent un moyen pratique pour l'interprétation des réponses des usagers comme le suggère le rapport QUALINET [102].
 - **Matériel de test** : Un ensemble de dix types de vidéos représentant différents domaines (sport, musique, dessins animés, etc.) sont utilisés. Les vidéos sont encodées en utilisant le même codec "AVC/H.264". Chaque vidéo a un degré de mouvement spécifique et des résolutions différentes (144p, 244p, 360p, 480p). La Figure (2.7) présente des captures d'écran des vidéos utilisées.



FIGURE 2.7 – Captures d'écran des vidéos utilisées lors de la campagne crowdsourcing

Le Tableau (2.2) résume les conditions générales des tests utilisées lors de l'évaluation de la QoE.

Paramètres	Description
Vidéos	-10 Types : Journal, Sport, Musique, etc. -Durée : 60 secondes -Résolution : 144p, 240p, 360p et 480p
Afficheurs	Samsung 5 : V. 4.2.2, SDK=19 Samsung 4 Mini : V. 4.2.2, SDK=19 Samsung 3 : V. 4.1.2, SDK=16 HTC X : V. 4.1.1, SDK=16 Archos Tablet : V. 4.0.4, SDK=16
Opérateurs	Orange, Free et Bouygues

TABLE 2.2 – Conditions générales des tests

La campagne de test a duré plus de 6 mois. Le panel de testeurs est composé de 155 hommes et 26 femmes. Une session de test dure au moins 5 minutes. 28 testeurs ont regardé au moins 10 vidéos sur une session.

À la fin de la campagne de test, la base de données récoltée [95] contient 1650 exemples avec 19 paramètres de différentes classes. La première classe concerne les paramètres réseau (QoS), qui sont collectés à partir des mesures des signaux reçus au niveau du terminal de test. La deuxième classe de paramètres concerne les facteurs applicatifs ("QoA" dans la Figure (1.2)). Elle désigne les indicateurs que fournit le lecteur vidéo VLC sur les caractéristiques de la séquence vidéo. La troisième classe concerne les informations spécifiques des terminaux utilisés pour les tests ("QoD" dans la Figure (1.2)).

2.5 Discussion

Dans ce chapitre, nous avons présenté deux campagnes d'évaluation que nous avons réalisées pour évaluer la QoE usager de service vidéo. La première concerne un environnement contrôlé réalisé dans le laboratoire de recherche LiSSI. La seconde campagne est réalisée sur le

terrain avec une campagne crowdsourcing (non-contrôlé).

Réaliser ce type d'évaluation requière beaucoup de temps et de moyens en plus de la mobilisation d'un grand panel de testeurs. Le Tableau (2.1) présente certaines bases de données existantes dans la littérature en accès libre. Celles-la concernent l'évaluation de la qualité perçue par l'utilisateur en termes de score MOS donnés par les utilisateurs de service vidéo.

En comparant nos propres bases de données [93, 95], par rapport aux bases de données existante (Tableau (2.1)), nous observons que nos bases de données contiennent un grand nombre de paramètres 19. De plus, ces deux bases de données disposent d'un large nombre d'exemples, avec 560 et 1650 exemples dans l'environnement contrôlé et non-contrôlé respectivement.

En outre, nous constatons que notre deuxième base de données [95] est plus fournie que toutes les autres dans le Tableau (2.1). Cela est très utile dans toute étude menant à la construction de modèles basés sur l'apprentissage automatique pour l'estimation de la QoE usager, où la disponibilité d'un grand échantillon améliore les performances du modèle prédictif. Pour finir, nous proposons ci-dessous quelques suggestions pour l'amélioration de la conduite des tests subjectifs et pour une meilleure estimation de la QoE.

- Utiliser des plateformes de crowdsourcing comme Microworkers ¹⁴.
- Inclure des questions de vérification dans le questionnaire. Ceci peut être réalisé en demandant aux testeurs par exemple de répondre par une valeur spécifique à un stimulus en affichant un message " *Pour cette séquence vidéo, répondez '1'* ". Le but est de s'assurer que les testeurs sont bien concentrés lors de la réalisation des tests.
- Re-contrôler les réponses de testeurs en utilisant les mêmes conditions d'essai pour différentes personnes.
- Utiliser des méthodes de nettoyage des données : supprimer les échantillons de données biaisés en utilisant des méthodes statistiques.

14. <https://microworkers.com/>

2.6 Conclusion

Les méthodes subjectives d'évaluation de la qualité vidéo sont bien connues et décrites. Elles sont utilisées pour construire des bases de données de la QoE en se basant sur de nombreuses normes et recommandations. Construire de telles bases de données est une tâche extrêmement difficile en raison de la mobilisation de participants humains, en plus du temps et des coûts financiers. Dans ce chapitre, nous avons présenté un certain nombre d'ensembles de données subjectives sur la QoE en premier lieu. Le but est d'obtenir une vue de l'état de l'art de l'évaluation de la QoE à l'aide de méthodes subjectives. En deuxième lieu, nous avons présenté nos deux campagnes de test, dans un environnement contrôlé et de crowdsourcing, que nous avons réalisées pour construire nos propres bases de données, qui nous serviront à évaluer la QoE usager d'un service vidéo.

Chapitre 3

Méthodes d'apprentissage automatique pour l'estimation de la Qualité d'Expérience usager

3.1 Introduction

Ce chapitre est notre première contribution dans le cadre de cette thèse. Il consiste à proposer une approche de sélection de facteurs, les plus pertinents, à utiliser pour évaluer la QoE usager en utilisant les méthodes d'apprentissage automatique. En premier lieu, nous présentons la problématique de sélection d'un sous-ensemble de caractéristiques pertinentes nécessaire au fonctionnement des méthodes d'apprentissage automatique. Ensuite, nous expliquons notre approche de sélection de facteurs à utiliser pour mieux estimer la QoE usager en termes de score MOS. Troisièmement, nous détaillons les méthodes d'apprentissage automatique utilisées dans les travaux de cette thèse. Pour finir, nous présentons une évaluation de ces méthodes d'apprentissage automatique et de notre approche de sélection de facteurs en utilisant la base de données présentée dans le chapitre 2 et qui est récoltée dans l'environnement contrôlé.

3.2 Sélection des meilleurs facteurs impactant la Qualité d'Expérience usager

3.2.1 Sélection des caractéristiques

Pour construire un modèle supervisé d'apprentissage automatique, plusieurs caractéristiques, appelées variables ou attributs, sont nécessaires comme entrées afin d'apprendre et de construire le modèle. Cependant, certaines études dans la littérature [64, 79] indiquent qu'une étape de sélection d'un sous-ensemble de caractéristiques pertinentes est nécessaire. Le but de cette étape est de rendre l'interprétation plus facile, de réduire le temps d'apprentissage et d'améliorer la généralisation du modèle en réduisant le sur-ajustement (réduction de la variance) [79]. Cela se traduit mathématiquement par la sélection d'un sous-ensemble F' (facteurs pertinents) de taille N' de tous les facteurs ($F = \{f_1, f_2, \dots, f_N\}$ de taille N). La fonction objective de cette modélisation est présentée comme suit :

$$Ev(F') = \max_{Z \subset F} Ev(Z) \quad (3.1)$$

Avec :

- Ev est la fonction qui permet d'évaluer un sous-ensemble de caractéristiques. L'objectif est que la plus grande valeur de Ev soit obtenue pour le meilleur sous-ensemble de caractéristiques (Z).
- $|Z| = N'$
- N' est soit un nombre prédéfini par l'utilisateur ou soit un nombre contrôlé par une des méthodes de génération de sous-ensembles.

3.2.2 Processus de sélection

Le processus de sélection des caractéristiques, des entrées des méthodes d'apprentissage automatique, est constitué de plusieurs composantes. D'après [11], ce processus inclut habituellement les quatre composantes résumées dans la Figure (3.1). Celle-ci illustre la décomposition modulaire de l'ensemble de ce processus.

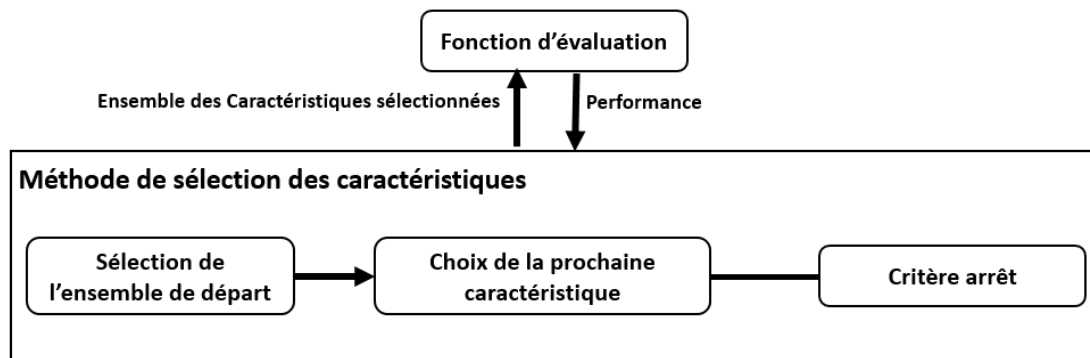


FIGURE 3.1 – Processus de sélection des caractéristiques

A) Sélection de l'ensemble de caractéristiques de départ :

- La première étape de sélection des caractéristiques consiste à fixer le point de départ de l'espace de caractéristiques (noté : F'_{init}). Une recherche peut commencer par un ensemble vide de caractéristiques, et continuer par l'ajout successif. Inversement, elle peut commencer avec l'ensemble de toutes les caractéristiques et continuer par la suppression séquentielle, à chaque itération. Une autre façon est de commencer la recherche par un sous-ensemble (F'_{init}) quelconque de caractéristiques[49].

B) Génération des prochaines caractéristiques :

- Une fois que l'ensemble de départ (F'_{init}) est choisi, une procédure de génération/recherche des futures caractéristiques à inclure dans F' est lancée. En général, les stratégies de recherche peuvent être classées en deux catégories :

B1) Génération exhaustive

Dans cette catégorie, une recherche exhaustive sur tous les sous-ensembles (F') de caractéristiques est effectuée afin de sélectionner le "meilleur" sous-ensemble de caractéristiques. Cette stratégie de recherche garantit de trouver le sous-ensemble optimal. Le problème majeur de cette approche est que le nombre de combinaisons croît exponentiellement en fonction du nombre de caractéristiques. Pour un ensemble de N

caractéristiques, et quand N devient grand, les 2^N combinaisons possibles rendent la recherche exhaustive impossible (problème NP-complet).

B2) Génération heuristique

Des algorithmes itératifs sont généralement utilisés dans cette catégorie. À chaque itération, l'algorithme permet de sélectionner ou de rejeter une ou plusieurs caractéristiques ($f_i \in F$). Les avantages de ces algorithmes sont leur simplicité et leur rapidité. En revanche, ils ne permettent pas de parcourir totalement l'espace de recherche. Dans la littérature, les trois sous-catégories les plus connues de cette approche sont :

- **Ascendante** : Commence avec un ensemble de caractéristiques vide ($F'_{init} = \emptyset$) et à chaque itération une ou plusieurs caractéristiques (f_i) seront ajoutées au fur et à mesure.
- **Descendante** : Commence avec l'ensemble de départ égale à l'ensemble total ($F'_{init} = F$) des caractéristiques et à chaque itération, une ou plusieurs caractéristiques seront supprimées.
- **Stepwise** : Consiste à ajouter ou supprimer des caractéristiques au sous-ensemble courant en mélangeant les deux approches précédentes.

C) Procédure d'évaluation des caractéristiques

- Dans la littérature, les méthodes utilisées pour évaluer un sous-ensemble de caractéristiques dans les méthodes d'apprentissage automatique peuvent être classées en trois catégories principales :

C1) Filtrage : Une évaluation basée sur la pertinence d'une caractéristique selon des mesures qui reposent sur les propriétés des données d'apprentissage indépendamment de la méthode d'apprentissage automatique utilisée. Le but de cette évaluation est de calculer un score pour évaluer le degré de pertinence de chacune des caractéristiques (x^i). Les mesures les plus utilisées pour le filtrage sont présentées dans le Tableau (3.1) ci-dessous :

Critère	Description	Calcul
Corrélation de Pearson	La corrélation de Pearson mesure la relation linéaire ("proportionnalité") entre les valeurs de deux variables.	$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[n \sum x^2 - (\sum x)^2][n \sum y^2 - (\sum y)^2]}}$ <p>Où : $x \in$ valeurs de la variable 1, $y_i \in$ valeurs de la variable 2 [72].</p>
Score de Fisher	Le critère de Fisher calcul le degré de séparabilité des classes (y_i) à l'aide d'une caractéristique donnée (x_i). On pourrait dire que la mesure est liée à la variance interclasse de la caractéristique.	$F(i) = \frac{\sum_{c=1}^C n_c (\mu_c^i - \mu^i)^2}{\sum_{c=1}^C n_c (\sigma_c^i)^2}$ <p>Où : n_c, μ_c^i et σ_c^i représentent respectivement l'effectif, la moyenne et l'écart type de la i^{me} caractéristique au sein de la classe c. μ^i est la moyenne globale de la i^{me} caractéristique.</p>
T-student	Le T-student permet de comparer les moyennes de deux groupes d'échantillons pour savoir si les moyennes des deux groupes sont significativement différentes au point de vue statistique.	$t(X_i) = \frac{(\mu_{i1} - \mu_{i2})}{\sqrt{\frac{\sigma_{i1}^2}{n_1} + \frac{\sigma_{i2}^2}{n_2}}}$ <p>Où : μ_{ij} représente la moyenne de la caractéristique X_i pour la classe C_j. σ_{ij} représente l'écart type de la i^{me} caractéristique X_i pour la classe C_j.</p>
Information mutuelle	L'information mutuelle mesure la dépendance entre les distributions de deux populations.	$I = \sum_{x_i} \sum_{y_i} P(X = x_i, Y = y_i) \log\left(\frac{P(X=x_i, Y=y_i)}{P(X=x_i)P(Y=y_i)}\right)$ <p>Où : μ_{ij} représente la moyenne de la caractéristique X_i pour la classe C_j. σ_{ij} représente l'écart type de la i^{me} caractéristique X_i pour la classe C_j.</p>
Vraisemblance	Le maximum de vraisemblance consiste à calculer la probabilité ou la densité d'un échantillon associée aux "n" données observées	<ul style="list-style-type: none"> - Déterminer la probabilité de "x" de l'ensemble d'apprentissage (T) : $p(x, T)$ ou $p(x C_k) // C_k$: La classe à prédire. - Déterminer le t optimal pour l'entrée de test x : $t = \text{argmax}\{ (x, T) \}$

TABLE 3.1 – Mesures utilisées par les méthodes de filtrage

C2) Enveloppantes : Les méthodes enveloppantes sont connues aussi par les méthodes "wrapper". Elles consistent à utiliser un modèle prédictif pour évaluer les sous-ensembles de caractéristiques. Dans cette évaluation, la base d'apprentissage est divisée en deux parties. La première est utilisée pour l'apprentissage et la deuxième pour valider la sélection des caractéristiques. Le sous ensemble ayant la meilleure performance, après

validation, est sélectionné à la fin.

C3) Intégrées : Les méthodes intégrées sont connues aussi par les méthodes "*embedded*".

Chaque méthode effectue la sélection de caractéristiques dans le cadre du processus de construction du modèle. Autrement dit, elle réalise simultanément la sélection des caractéristiques (x_i) et le calcul de la variable à prédire (y). Contrairement aux méthodes enveloppantes (*wrapper*), ces méthodes peuvent se servir de tous les exemples d'apprentissage pour construire l'algorithme de sélection des caractéristiques. Ce qui peut améliorer les résultats des méthodes d'apprentissage automatique.

D) Critère d'arrêt

- Pour arrêter le processus de sélection des caractéristiques, certains critères doivent être définis. Le critère choisi dépend de l'approche d'évaluation des caractéristiques utilisées. En effet, pour les méthodes de filtrage, le critère d'arrêt couramment utilisé est basé sur l'ordre des caractéristiques, rangées selon certains scores de pertinence (généralement des mesures statistiques). Une fois les caractéristiques ordonnées, celles qui ont les scores les plus élevés seront choisies et utilisées par un classificateur. Dans le cas des méthodes de type enveloppantes ("*wrapper*"), le processus de recherche peut s'arrêter lorsque il n'y a plus d'amélioration de précision. En d'autres termes, lorsqu'il n'y a plus la possibilité de trouver un sous-ensemble meilleur que le sous-ensemble actuel. Pour les méthodes intégrées ("*embedded*"), le processus s'arrête au moment où la précision dépasse un certain seuil défini par l'utilisateur.

3.2.3 Sélection des facteurs de la Qualité d'Expérience

Dans le cas de construction de modèles d'estimation de la QoE usager, par opposition aux méthodes objectives, les modèles basés sur les méthodes d'apprentissage automatique ont besoin de données qui sont issues de l'ensemble des composants de la chaîne de traitement et de transport (utilisateur, périphérique, application, des éléments du réseau). Ces données sont appelées ici "facteurs" (notés *QoE IFs*) et ils sont présentés et classifiés dans la Section (1.4).

Dans le cas du service de diffusion vidéo qui représente notre cas d'usage, il y a beaucoup de facteurs qui ont un impact sur la qualité de la vidéo perçue par l'utilisateur final (par exemple :

le débit binaire vidéo ou encore la fréquence d'images vidéo). Toutefois, un modèle qui estime le ressenti usager (QoE) sur la base d'une combinaison de ces facteurs peut produire de mauvais résultats sous certaines conditions comme expliqué dans la littérature [90, 16]. Cela est dû à ce que nous nous appelons "*problème d'interaction des facteurs*". Celui-ci peut provoquer un sur-ajustement (grande variance) et impacter négativement les performances de prédiction des méthodes d'apprentissage automatique supervisé [79]. Ainsi et pour mieux estimer la qualité perçue par l'utilisateur (QoE), nous devons donc sélectionner en amont les facteurs les plus pertinents qui seront utilisés par la suite dans le calcul de la QoE usager en termes de score MOS.

Dans la suite de ce chapitre, nous allons présenter en premier lieu les méthodes d'apprentissage automatique utilisées. En deuxième lieu nous allons détailler notre approche pour sélectionner les facteurs les plus pertinents.

3.3 Méthodes d'apprentissage automatique

Avant de présenter notre approche de sélection de facteurs, nous introduisons les méthodes d'apprentissage automatique utilisées pour calculer les scores MOS usager. 6 méthodes (2 méthodes classiques et 4 méthodes ensemblistes) sont ainsi implémentées et testées dans le cadre de cette thèse. Elles sont toutes présentées ci-dessous :

3.3.1 Méthodes classiques

(A1) Machines à vecteurs de support

Les machines à vecteurs de support, ou plus connues sous l'appellation anglaise "*Support Vector Machines (SVM)*", est une méthode d'apprentissage supervisé. Son principe de fonctionnement consiste à identifier les différentes classes (positive, négative) en déterminant l'hyperplan de séparation maximale entre les deux classes à prédire. Selon la nature de l'ensemble de données d'apprentissage, différents noyaux peuvent être utilisés pour construire un SVM. Cela s'explique par le fait que c'est le noyau qui détermine la forme de l'objet séparant l'hyperplan. Par exemple, un noyau linéaire peut être utilisé dans les cas où les données sont linéaires et séparables en fournissant une équation linéaire pour représenter l'hyperplan. Cependant, d'autres

noyaux sont nécessaires dans les cas où les données ne sont pas linéairement séparables. Ces noyaux permettent de résoudre le problème non-linéaire par une projection implicite dans un espace de dimension supérieure.

L'un des noyaux les plus utilisés est le noyau gaussien (que nous avons utilisé). Celui-ci modélise les points de données à partir de leur espace d'entrée d'origine dans un espace d'éléments de haute dimension. La sortie du SVM avec Le noyau gaussien (aussi connu sous le nom de *SVM-RBF*) est présentée par l'équation (3.3) [113] :

$$f(x) = w^T \Phi(x) + b \quad (3.2)$$

Où : $\Phi(x)$ représente le noyau utilisé. L'objectif est de déterminer le vecteur poids w^T et d'intercepter b qui minimise les fonctions objectives suivantes :

$$\min_{(b,w)} \frac{1}{2} w^2 + C \sum_{i=1}^m [(y_i) \times cost_1(f(x_i)) + (1 - y_i) \times cost_0(f(x_i))] \quad (3.3)$$

Où : C est un paramètre de régularisation qui pénalise les instances mal classées, $cost_i$ est l'erreur quadratique sur l'ensemble des données d'entraînement.

Par ailleurs, les SVMs sont aussi utilisés pour la régression comme l'indique l'étude [58]. En effet, le principe de fonctionnement consiste à trouver la fonction $f(x)$ qui a au plus une déviation ε par rapport aux exemples d'apprentissage (x_i, y_i) , pour $i = 1, \dots, N$, et qui est la plus plate possible. Cela revient à ne pas considérer les erreurs inférieures à ε et à interdire celles supérieures à ε .

(A2) Réseaux de neurones aléatoire

Les réseaux de neurones sont des outils mathématiques principalement utilisés pour les tâches de classification. Dans ces réseaux, un vecteur x d'entrée est soumis aux neurones qui retournent en sortie un vecteur de sortie " s " correspondant à une décision de classification de x dans une classe donnée. L'élément constitutif de base d'un réseau de neurones artificiel est le perceptron ou plus connu sous l'appellation "neurone formel". Celui-ci est un processus élé-

mentaire réalisant une fonction $\Phi : K^n \rightarrow K$ (avec $K = R$ (réel) ou B (booléen)). Pour cela, il procède en deux étapes :

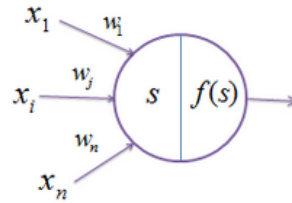


FIGURE 3.2 – Neurone formel

1/ il calcule son état d'excitation interne s en fonction de ses entrées (x_i) et des poids (w_i) associés à chaque entrée ;

2/ il émet un résultat $y = f(\sigma)$ en sortie, selon son état d'excitation. La fonction f est appelée la fonction d'activation. Les fonctions d'activation les plus utilisées sont : la fonction échelon, la fonction signe et la fonction sigmoïde (Figure (3.3)).

$$\begin{aligned} \text{fonction échelon : } f(\sigma) &= \begin{cases} 1 & \text{si } \sigma \geq 0 \\ 0 & \text{si } \sigma < 0 \end{cases} \\ \text{fonction signe : } f(\sigma) &= \begin{cases} 1 & \text{si } \sigma \geq 0 \\ -1 & \text{si } \sigma < 0 \end{cases} \\ \text{fonction sigmoïde : } f(\sigma) &= \frac{1}{1+e^{-\sigma}} \end{aligned}$$

FIGURE 3.3 – Fonctions d'activation

Par construction, un réseau de neurones constitue un assemblage de neurones formels élémentaires (Figure (3.2)). Deux types de neurones peuvent être distingués :

- les neurones visibles dont la valeur de sortie est observable ;
- les neurones cachés dans les couches intermédiaires et dont l'état interne est inconnu. Leur(s) sortie(s) est(sont) connectée(s) à d'autres neurones cachés ou aux neurones de sortie.

Enfin, l'objectif des réseaux de neurones consiste à l'approximation de n'importe quelle fonction non-linéaire en jouant sur les poids (w_i) qui constituent des paramètres de la fonction

d'activation utilisée (f). L'apprentissage offre ainsi un moyen efficace pour calculer ces poids à partir d'exemples (x_i, y_i) connus de valeurs de cette fonction[32].

- Le Tableau (3.2) résume les configurations utilisées par les méthodes classiques d'apprentissage automatique pour calculer la QoE usager en termes de scores MOS.

Description de la méthode	Configurations
<p>☐ RNN :</p> <p>▶ hidden : Nombre de neurones par couche (h).</p> <p>▶ threshold : Erreur de dérivées partielles (th).</p>	<p>$h \in \{2, 3, 4, 5, 6\}$</p> <p>$th : 0.1 \text{ à } 0.9$</p>
<p>☐ SVM :</p> <p>▶ type : Classification or régression.</p> <p>▶ kernel : Type du noyau.</p> <p>▶ γ : Inverse du rayon d'influence des échantillons sélectionnés.</p> <p>▶ nu : La limite inférieure et supérieure sur le nombre d'exemples qui se trouvent du mauvais côté de l'hyperplan.</p>	<p>$type = regression$</p> <p>$kernel \in \{radial\}$</p> <p>$\gamma : 0.1 \text{ à } 0.9$</p> <p>$nu : 0.1 \text{ à } 0.9$</p>

TABLE 3.2 – Configurations utilisées par les méthode classiques de l'apprentissage automatique

3.3.2 Méthodes ensemblistes

Les méthodes ensemblistes ont été proposées pour contrer l'instabilité des arbres de décisions et améliorer leurs performances prévisionnelles. Une méthode ensembliste est définie comme un ensemble de classificateurs individuels de différents types, dont les décisions sont combinées par moyenne ou vote pour donner une décision plus précise. Deux types de méthodes ensemblistes existent dans la littérature : les méthodes *bagging* et les méthodes *boosting*. Ci-dessous leur principe de fonctionnement :

(B1) Bagging :

Le Bagging est un acronyme de *Bootstrap Aggregating*. Il s'agit d'un moyen de réduire la variance de la prédiction en générant plusieurs sous-ensembles d'apprentissage à partir du

jeu de données original. Son fonctionnement consiste à entraîner plusieurs classifieurs ("N") sur ces sous-ensembles constitués de $2/3$ des données. Le résultat de prédiction est obtenu en faisant la moyenne des réponses des "N" classifieurs ou par vote majoritaire. Dans le cadre de la thèse, nous avons choisi d'utiliser deux méthodes bagging qui sont : les forêts aléatoires, plus connu sous l'appellation anglaise Random Forest (*RF*) [37, 103] et le bagging à base d'arbres de décisions noté *baggingDT* [36, 133].

(B2) Boosting :

Le Boosting est une méthode qui consiste à générer des classificateurs faiblement décorrélés au lieu d'entraîner un seul classificateur complexe et la prédiction est obtenue en pondérant les réponses de tous les classificateurs. L'idée maîtresse est de combiner plusieurs classificateurs afin d'en obtenir un plus performant.

Avec le boosting, contrairement au bagging, un poids différent est associé à chaque observation, et il est successivement modifié en fonction du nombre de fois où l'observation a été mal classifiée par les classificateurs construits précédemment. Plus l'observation a été mal classifiée, plus le poids qui y est associé sera élevé. Le but est de forcer les classificateurs subséquents à se concentrer sur les données les plus difficile à classifier. Par conséquent, le classificateur k dépend de la performance des $(k - 1)^{me}$ classificateurs. Le classificateur final est construit par un vote pondéré de tous les classificateurs, chacun étant pondéré par rapport à sa précision sur l'ensemble avec lequel il a été entraîné

Dans le cadre de la thèse, nous avons choisi d'utiliser deux méthodes boosting. La première méthode est le boosting à base des arbres de décisions [25, 56], noté *BoostDT* et plus connu aussi sous le nom de *AdaBoost*. La deuxième méthode est le boosting à base du gradient noté *BoostGrad* [61, 141].

Le Tableau (3.3) résume les configurations utilisées par les méthodes ensemblistes de l'apprentissage automatique :

Description de la méthode	Configurations
<input type="checkbox"/> RF : ► n : Nombre de forêts aléatoire utilisé	$n \in \{10, \dots, 100, 200\}$
<input type="checkbox"/> BaggingDT : ► Type : Méthode utilisée.	$type = Decision\ Trees$
<input type="checkbox"/> BoostDT (Blackboost) : ► mstop : Nombre maximal d'arbres de décision.	$mstop = 10000$
<input type="checkbox"/> BoostGrad : ► Type : Méthode utilisée. ► family : La famille de méthode utilisée	$type = gradient$ $family = "gaussian"$

TABLE 3.3 – Configurations utilisées par les méthode ensemblistes

3.3.3 Discussion

Dans cette partie, nous présentons une étude comparative des méthodes d'apprentissage automatique que nous avons utilisées dans le cadre de cette thèse. La Figure (3.4) illustre les critères utilisés lors de la comparaison.

Contrairement aux méthodes classiques d'apprentissage automatique (réseaux de neurones, arbres de décision, machines à vecteurs de support), les méthodes ensemblistes (Adaboost, les forêts aléatoire, etc.) permettent d'améliorer la précision surtout dans le cas d'utilisation d'un grand échantillon de données d'apprentissage. Selon [188], le boosting et les forêts aléatoires sont celles qui permettent d'obtenir les meilleures performances. Cependant, les méthodes ensemblistes ont l'inconvénient de faire disparaître l'avantage des méthodes d'apprentissage automatique classique qui est la facilité d'interprétation (comme les réseaux de neurones). Les prochaines années verront probablement l'émergence de nouvelles méthodes ensemblistes donnant des résultats de plus en plus précis tout en étant interprétables.

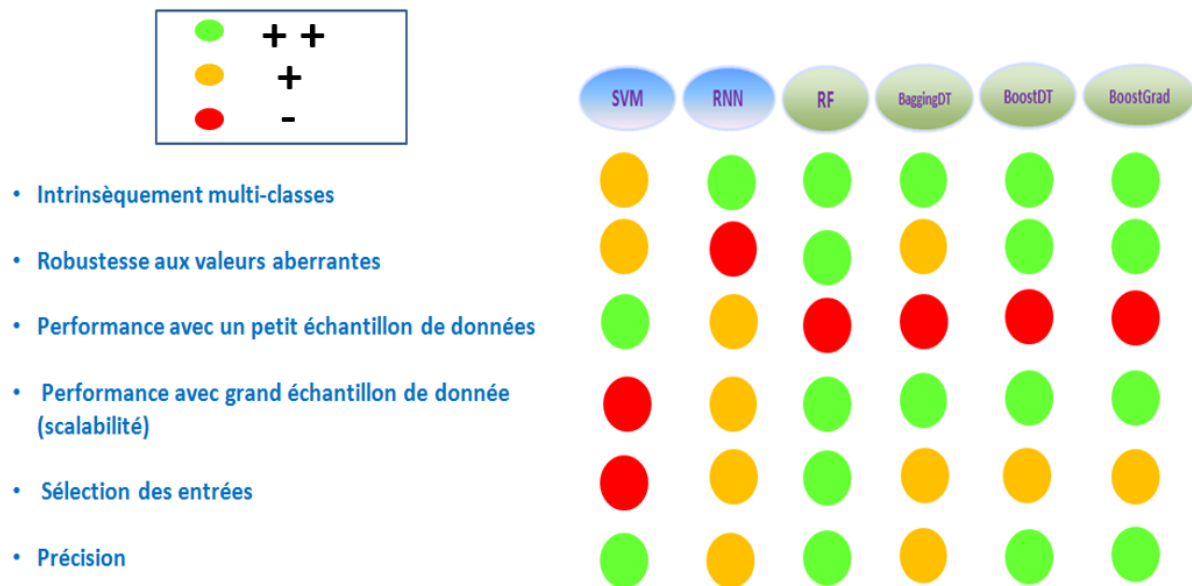


FIGURE 3.4 – Comparaison des méthodes d’apprentissage automatique

3.4 Estimation de la Qualité d’Expérience avec l’apprentissage automatique

Nous avons utilisé 6 méthodes d’apprentissage automatique pour estimer la perception usager (QoE). Ces méthodes sont : les machines à vecteurs de support (SVM)[113], les réseaux de neurones (RNN)[32], les forêts aléatoires (RF)[37], le bagging basé sur les arbres de décisions (BaggingDT) [36], le boosting basé sur les arbres de décision (BoostDT) [25] et le boosting basé sur le gradient (BoostGrad)[61]. Chaque méthode donne une valeur calculée de la QoE en termes de score MOS (noté $MOS_{pred_i}^{t_k}$). Pour déterminer celle-ci, deux étapes sont nécessaires pour chaque méthode : le calcul et la validation.

Pour le calcul des valeurs prédites, diverses configurations sont utilisées par chaque méthode d’apprentissage automatique comme indiqué dans les Tableaux (3.2) et (3.3). Les meilleures valeurs $MOS_{pred_i}^{t_k}$ sont ainsi sélectionnées en fonction de la meilleure configuration de chaque méthode par rapport à un indicateur de performance. Dans les travaux de cette thèse, nous

avons opté pour l'utilisation de l'erreur absolue moyenne (notée "*MAE*") comme le suggère [6, 158, 19], qui explique que cet indicateur, dans certains cas, peut être un meilleur indicateur que l'erreur quadratique (*RMSE*). La formule de calcul de l'erreur absolue moyenne (*MAE*) est présentée dans l'équation (3.4),

$$MAE = \frac{1}{n} \sum_1^n |MOS_{pred_i} - MOS_{real_i}| \quad (3.4)$$

Où : MOS_{pred_i} les valeurs de score MOS prédites, MOS_{real_i} les valeurs de score MOS réelles et n le nombre total d'échantillons.

Par exemple, plusieurs valeurs du nombre de forêts (configurations : 10, 20, 30, 40, 50, 100, 200) sont utilisées dans le cas de la méthode RF (Tableau (3.3)). Nous comparons ces configurations en calculant le taux d'erreur absolue moyenne.

À la fin, nous constatons que l'utilisation de 100 forêts nous permet d'avoir le minimum taux d'erreur absolue moyenne (*MAE*) le plus faible. Par conséquent, nous considérons cette valeur (forêts de 100) comme la meilleure configuration à utiliser dans la méthode "RF".

Pour valider les résultats, la technique de validation croisée est utilisée. Le processus suivi est présenté dans la Figure (3.5) et il se résume comme suit : 9/10 de données subjectives sont utilisées pour la construction du modèle (apprentissage) et le reste pour le test. Ce processus est répété dix fois pour valider les résultats ($n = 10$).

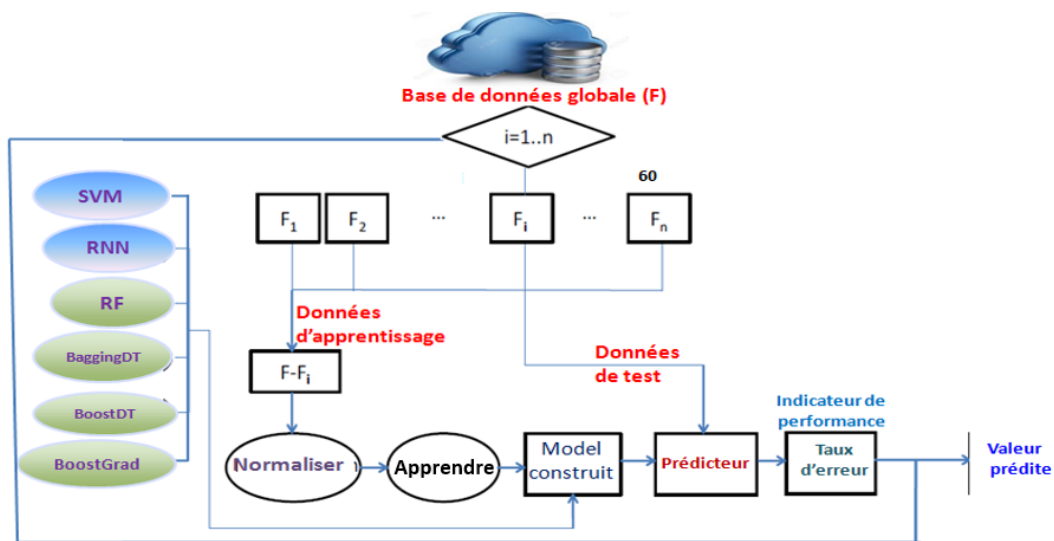


FIGURE 3.5 – Validation croisée

3.5 Approche de sélection de facteurs impactant la QoE usager

3.5.1 Présentation

Nous proposons ici une nouvelle approche qui permet de sélectionner les facteurs les plus pertinents (F') lors de l'évaluation de la QoE usager en utilisant les méthodes d'apprentissage automatique. Le processus de sélection de facteurs inclut quatre composantes résumées dans la Figure (3.1). Dans notre approche, ces composantes sont présentées dans la Figure (3.6).

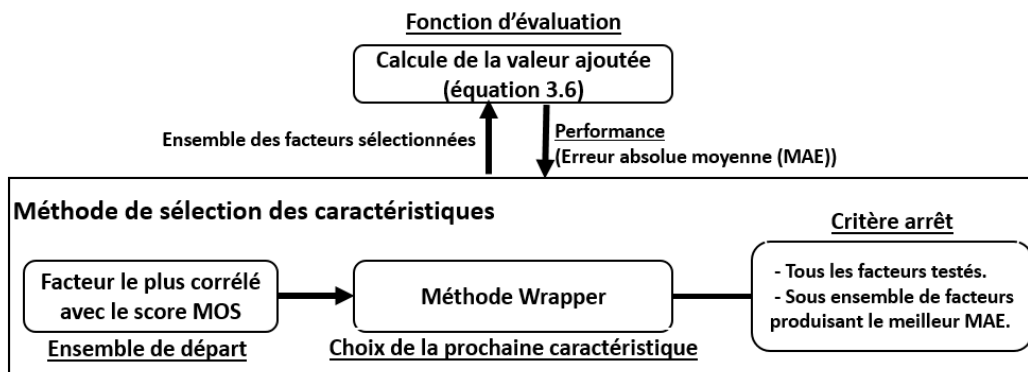


FIGURE 3.6 – Approche proposée pour sélectionner les facteurs pertinents

Cette approche consiste à utiliser une méthode ascendante pour explorer les sous-ensembles des facteurs (F), que nous initions (F_{init}) au début, en utilisant le coefficient "r" de Pearson[73]. Celui-ci est utilisé à cause de sa simplicité de calcul et ses performances d'analyse des relations linéaires entre des ensembles de données (détecte automatiquement la modification de l'une des variables qui est associée à une modification proportionnelle de l'autre variable). Dans l'approche proposée, le facteur le plus corrélé avec le score MOS des usagers est utilisé comme ensemble de départ (F_{init}).

Concernant la sélection de facteurs, une fonction d'évaluation a été définie (équation 3.6) et une méthode enveloppante ("*wrapper*") a été utilisée. Celle-ci se base sur l'usage des méthodes d'apprentissage automatique pour évaluer les sous-ensembles de caractéristiques selon l'erreur absolue moyenne (MAE). La technique du retour arrière (*back-jumping*) est utilisée aussi pour choisir les facteurs les plus bénéfiques selon la/les meilleure(s) méthode(s) d'apprentissage automatique.

Enfin, deux critères d'arrêt permettent d'arrêter notre processus de sélection, soit quand tous les facteurs sont sélectionnés ou soit quand un sous-ensemble de ces facteurs est sélectionné et que les autres facteurs ont tous été testés, avec le retour arrière (*back-jumping*), sans qu'ils apportent une amélioration.

3.5.2 Formulation mathématique

L'idée maîtresse de notre approche de sélection de facteurs QoE consiste à calculer la valeur ajoutée de chaque facteur et à tester si sa combinaison avec les facteurs déjà sélectionnés (notés : *QoEIFs*) améliore la précision d'estimation de(s) la(es) méthode(s) d'apprentissage automatique.

Pour commencer, on fixe d'abord l'ensemble de départ des facteurs sélectionnés (F_{init}). Pour cela, on utilise la matrice de corrélation (M), entre les facteurs et le score MOS usager (QoE). Le premier facteur sélectionné est celui ayant la plus grande valeur de corrélation avec le MOS.

Pour le reste des facteurs (à l'exception du facteur le plus corrélé), posant " f_candi " tout facteur non sélectionné et candidat à l'être dans les prochaines étapes. L'ensemble de ces facteurs est noté : " F_{test} ". L'idée de notre approche est de commencer par classifier tous les $F_{test} = \{f_candi\}$ dans une file " F " en calculant leur valeur ajoutée à l'aide d'une fonction d'utilité. Celle-ci est définie dans l'équation (3.5) et donnée en (3.6) :

$$f_benefit(x, f_used) : [-1, 1] \times [-1, 1] \longrightarrow [0, 2] \quad (3.5)$$

$$f_benefit(x, f_used) = (1 - AV(r(x, f_used))) + r(AV(x), MOS) \quad (3.6)$$

Où $f_benefit$ est la fonction qui quantifie la valeur ajoutée du facteur x . La variable x représente un des facteurs candidats à être sélectionné (f_candi), $\mathbf{r}(\mathbf{x}, \mathbf{y})$ représente la valeur de la corrélation ' r ' de Pearson de la matrice M entre (\mathbf{x}, \mathbf{y}) et $AV(\mathbf{x})$ représente la valeur absolue de la variable \mathbf{x} .

Justification

- L'utilisation de la valeur 1, est justifié par l'utilisation de la valeur absolue (AV) de la corrélation " r " de Pearson ($AV(r) \in [0, 1]$).
- $(1 - AV(r(f_candi, f_used)))$ représente l'impact (interaction/répulsion) entre les facteur candidat (f_candi) et le dernier facteur sélectionné (f_used).
- $r(AV(f_candi), MOS)$ représente l'interaction entre le facteur candidat (f_candi) et le score MOS.

Dans le cas où le premier facteur candidat (tête de " F ") n'améliore pas le taux d'erreur global, le deuxième facteur de " F " est testé en utilisant la technique *retour arrière* (*back-jumping*). Le processus se répète jusqu'à atteindre un des critères d'arrêts. Soit quand tous les facteurs de " F_{test} " sont sélectionnés ou soit quand un sous-ensemble de ces facteurs est sélectionné et les autres facteurs ont été tous testés avec le retour arrière (*back-jumping*) et qu'aucune amélioration n'a pu être trouvée.

Le diagramme présenté dans la Figure (3.7) décrit toutes les étapes de la démarche utilisée pour choisir les facteurs les plus pertinents. Le processus de sélection de facteurs peut se résumer comme suit :

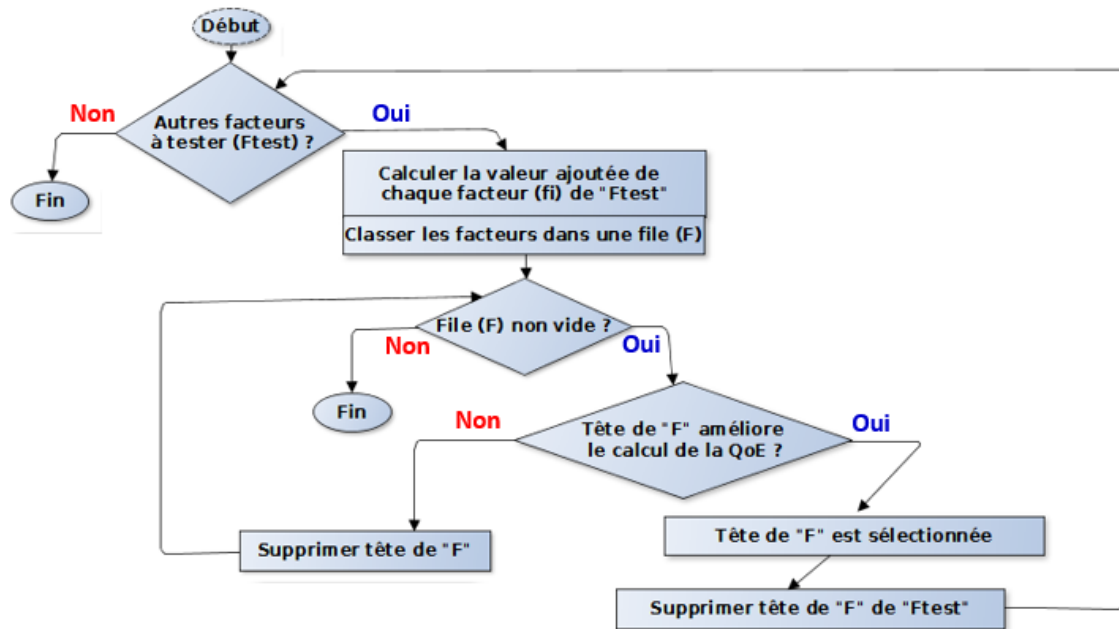


FIGURE 3.7 – Diagramme des étapes de la sélection de facteurs

- Le premier facteur sélectionné est le facteur ayant la plus grande valeur de corrélation ("r") avec le MOS usager dans la matrice de corrélation (M).
- Au début de chaque étape, le taux d'erreur absolue moyenne (MAE_i) est calculé en ajoutant le facteur (f_candi) ayant la plus grande valeur rajoutée. Le but est de vérifier si la valeur courante du taux d'erreur (MAE_i) (en rajoutant f_candi aux facteurs déjà sélectionnés (f_used)) est inférieure à la valeur de l'erreur globale de l'étape précédente (MAE_{i-1}).
- À chaque étape, si le taux actuel d'erreur est supérieur ($MAE_i > MAE_{i-1}$) en rajoutant la tête de la file (F), les facteurs suivants sont testés en utilisant la technique du retour

arrière (*back-jumping*). Cela est réalisé selon l'ordre dans la file " F " pour observer si un de ces facteurs diminue le taux d'erreur.

- Le processus de sélection de facteurs s'arrête lorsque (i) tous les facteurs (F) seront sélectionnés, ou (ii) aucun des facteurs non sélectionné n'améliore la prédiction (MAE_i) par rapport à l'étape précédente MAE_{i-1} .

3.6 Évaluation de l'approche de sélection de facteurs

3.6.1 Description

Cette section présente une évaluation de notre approche de sélection de facteurs les plus pertinents de la QoE. Cette évaluation se base sur l'utilisation des méthodes d'apprentissage automatique présentées dans la Section (3.3). Ces méthodes sont les machines à vecteurs de support (SVM)[113], les réseaux de neurones (RNN)[32], les forêts aléatoires (RF)[37], le bagging basé sur les arbres de décisions (BaggingDT) [36], le boosting basé sur les arbres de décision (BoostDT) [25] et le boosting basé sur le gradient (BoostGrad)[61]. La précision de notre approche est évaluée en utilisant la base de données collectée [93] lors de la campagne de test en environnement contrôlé (Chapitre (2)).

Dans notre évaluation, les 11 facteurs du niveau applicatif sont utilisés (c.f Tableau (3.4)). Ils sont récupérés au niveau du lecteur vidéo VLC comme des indicateurs. Le choix de ces facteurs est justifié par le grand impact qu'ils ont sur le ressenti final de l'utilisateur et par leur collecte qui se fait d'une façon non intrusive au niveau du terminal du client[116, 20]. Le Tableau (3.4) détaille ces facteurs. Cette évaluation est effectuée sous l'outil R software[1] où toutes les méthodes d'apprentissage automatique (Section (3.3)) ont été testées avec plusieurs configurations pour sélectionner celle qui donne la meilleure précision comme présenter dans la Section (3.4).

Facteur	Sym	Description
Vidéo id	f_{CTN}	Entier entre 1 et 8 présentant le type de la vidéo (journal, musique, sport, etc.).
Mouvement	f_{MVT}	Degré des mouvements durant la séquence vidéo (peu, moyen or beaucoup).
Résolution	f_{Resol}	Résolution affichée (144p, 240p,..., 1080p (full HD)).
Vlc_caching	f_{Cach}	Valeur du tampon (cache) dans le lecteur vidéo VLC (ms).
Taille vidéo	f_{Size}	Nombre total des paquets téléchargés.
Bitrate	f_{BR}	Valeur du débit binaire vidéo transmis ou traités par seconde.
Frame rate	f_{FR}	Nombre d'images affichées par seconde.
Frame lost	f_{FL}	Nombre d'images perdues par seconde.
Audio rate	f_{AR}	Valeur du débit binaire audio transmis ou traités par seconde.
Audio lost	f_{AL}	Nombre de paquets audio perdus par seconde.
Buffer	f_{Buff}	Temps initial de mise en tampon (ms).
MOS	mos	Valeur du MOS (Mean Opinion Score).

TABLE 3.4 – Description des paramètres vidéo utilisés

3.6.2 Processus

Notre processus d'évaluation est résumé comme suit : Nous commençons par construire la matrice de corrélation (M) de tous les facteurs et le score MOS usager à partir de la base de données [93](Figure (3.8)). Cette matrice représente les relations linéaires entre les différents facteurs et avec le score MOS usager. Elle se base sur le calcul de la valeur du corrélation "r" de Pearson [73]. Celle-ci représente une valeur réelle entre (-1) et $(+1)$ qui détecte automatiquement la modification de l'un des facteurs qui est associée à une modification proportionnelle de l'autre facteur. Le coefficient "r" est mesuré comme suit :

- Si les deux facteurs testés ont tendance à augmenter ou à diminuer ensemble, le coefficient "r" est positif (tend vers $(+1)$ comme la relation entre le nombre d'images par secondes (f_{FR}) et le débit binaire vidéo (f_{BR}) : $r(f_{FR}, f_{BR})=0.35$).

f_{CT}	•	•	•	•	•	•	•	•	•	•	•	•	•	•
0.03	f_{MVT}	•	•	•	•	•	•	•	•	•	•	•	•	•
-0.06	-0.1	f_{Resol}	•	•	•	•	•	•	•	•	•	•	•	•
-0.01	0.05	-0.02	f_{Cach}	•	•	•	•	•	•	•	•	•	•	•
0.07	-0.01	0.29	0.11	f_{Size}	•	•	•	•	•	•	•	•	•	•
0.01	-0.09	0.69	-0.06	0.32	f_{BR}	•	•	•	•	•	•	•	•	•
0.06	-0.01	0.32	0.12	0.92	0.35	f_{FR}	•	•	•	•	•	•	•	•
0	-0.02	-0.16	-0.14	-0.22	-0.16	-0.48	f_{FL}	•	•	•	•	•	•	•
0.11	-0.05	0.18	0.15	0.82	0.2	0.86	-0.28	f_{AR}	•	•	•	•	•	•
0.04	-0.04	0.26	-0.05	0.19	0.24	0.07	0.19	0.02	f_{AL}	•	•	•	•	•
-0.07	-0.05	0.02	-0.14	0.24	0.02	0.17	0.11	0.23	0.16	f_{Buff}	•	•	•	•
0.13	-0.07	0.49	0.08	0.63	0.51	0.7	-0.36	0.68	-0.08	0.04	MOS	•	•	•

FIGURE 3.8 – Matrice de corrélation de 'r' Pearson

- Si une variable a tendance à augmenter lorsque l'autre diminue, le coefficient est négatif (tend vers -1) comme la relation entre le score MOS (MOS) et le taux de perte d'images par seconde (f_{FL}) : $r(f_{FR}, f_{BR}) = -0.36$.

Les étapes du processus de sélection de facteurs sont ensuite exécutées comme suit :

Étape 1 : La matrice de corrélation M (Figure 3.8) obtenue à l'étape initial est utilisée de la façon suivante pour initialiser les variables utilisées au départ :

- ▶ $f_{used} = f_{FR}$ // Facteur ayant la plus grande valeur de corrélation par rapport au MOS.
- ▶ $F_{init} = \{f_{FR}\}$
- ▶ $F_{test} = \{f_{CT}, f_{MVT}, f_{Resol}, f_{Cach}, f_{Size}, f_{BR}, f_{FL}, f_{AR}, f_{AL}, f_{ABuff}\}$

Étape 2 : On exécute le processus de sélection de facteurs présenté à la Section (3.5.2) afin

de réduire le taux d'erreur. Dans cette étape, le taux moyen d'erreur absolue est de 0.428 en utilisant la méthode boosting à base du gradient ("BoostGrad").

S	Facteurs	Variante 1	MAE	Meilleure méthode AA
2	$f_{FR} f_{Resol}$	$(1 - 0.32) + 0.49 = 1.17$	0.428	Boosting gradient

Étape 3 : Dans la troisième étape, nous avons rajouté un nouveau facteur " f_{AR} " (débit binaire audio) à ceux déjà sélectionnés, ce qui a permis de baisser le taux d'erreur pour atteindre 0.346 en utilisant la méthode forêts aléatoires ("RF").

S	Facteurs	Variante 2	MAE	Meilleure méthode AA
3	$f_{FR} f_{Resol} f_{AR}$	$(1 - 0.18) + 0.68 = 1.53$	0.346	Forêts aléatoires

Étape 4 : Dans cette étape, le facteur " f_{BR} " (débit binaire vidéo) est sélectionné. À l'image de l'étape 3, ce facteur est rajouté aux facteurs déjà sélectionnés, ce qui a permis de réduire encore le taux d'erreur pour atteindre 0.282 en utilisant la méthode forêts aléatoires ("RF").

S	Facteurs	Variante 3	MAE	Meilleure méthode AA
4	$f_{FR} f_{Resol} f_{AR} f_{BR}$	$(1 - 0.2) + 0.51 = 1.31$	0.252	Forêts aléatoires

Étape 5 : Dans la cinquième étape, l'ajout du premier facteur de la file "F" (f_{Size}) ne permet pas de diminuer le taux d'erreur de l'étape précédente. Dans ce cas, le deuxième facteur de la file "F" est testé, en utilisant la technique du retour arrière (*back-jumping*), pour avoir un nouveau taux d'erreur MAE_5 égale à 0.227 plus faible en utilisant la méthode boosting qui se base sur le gradient ("BoosGrad").

S	Facteurs	Variante 4	MAE	Meilleure méthode AA
5	$f_{FR} f_{Resol} f_{AR} f_{BR} f_{Size}$	$(1 - 0.22) + 0.63 = 1.41$	0.261	Boosting gradient
	$f_{FR} f_{Resol} f_{AR} f_{BR} f_{FL}$	$(1 - 0.16) + 0.51 = 1.35$	0.244	Boosting gradient

Étape 6 : Dans l'étape 6, aucune amélioration des performances de calcul n'est trouvée comme indiqué par le résultat ci-dessous (après avoir fait plusieurs retours arrière) :

S	Facteurs	Variante 5	MAE	Meilleure méthode AA
6	$f_{FR} f_{Resol} f_{AR} f_{BR} f_{FL} f_{Size}$	1.31	0.259	Boosting gradient
	$f_{FR} f_{Resol} f_{AR} f_{BR} f_{FL} f_{CT}$	1.12	0.263	
	$f_{FR} f_{Resol} f_{AR} f_{BR} f_{FL} f_{ABuff}$	1.06	0.262	
	$f_{FR} f_{Resol} f_{AR} f_{BR} f_{FL} f_{Cach}$	1.02	0.256	
	$f_{FR} f_{Resol} f_{AR} f_{BR} f_{FL} f_{AL}$	0.98	0.265	

L'évaluation réalisée précédemment illustre les étapes nécessaires pour sélectionner les meilleurs facteurs. Dans les premières étapes de notre approche (étapes 2, 3 et 4), le facteur ayant la plus grande valeur ajoutée (calculée avec " f_{benefi} " (équation 3.6)) est toujours sélectionné.

Dans l'étape 5, le premier facteur classifié dans " F " ne donne plus de bon résultat en raison du problème d'interaction entre les facteurs [90, 16]. C'est pourquoi, le facteur ayant la deuxième grande valeur ajoutée (calculée avec " f_{benefi} " (équation 3.6)) est sélectionné, suite à l'utilisation de la technique retour arrière.

Dans l'étape 6, aucun facteur candidat ne donne un taux d'erreur plus faible que l'étape précédente (MAE_5 qui est égale à 0.244 avec la méthode boosting basée sur le gradient ($BoostGrad$)). La Figure (3.9) résume l'exécution du processus de sélection des meilleurs facteurs en utilisant la base de données récoltée lors de la campagne d'évaluation dans le laboratoire [93].

À la fin du processus de sélection, notre approche a sélectionné cinq facteurs (f_{Resol} ,

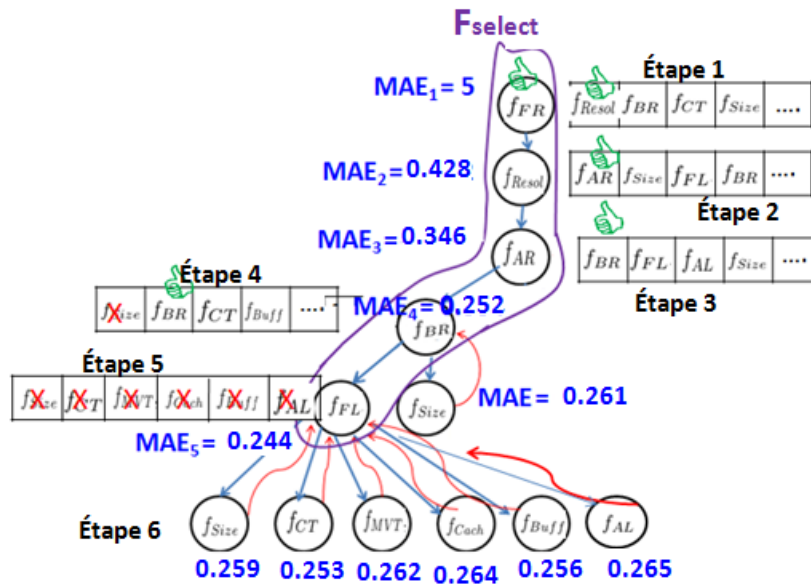


FIGURE 3.9 – Exécution du processus de sélection des facteurs

f_{FR} , f_{FL} , f_{BR} et f_{AR}) et la meilleure méthode d'apprentissage automatique résultante est la méthode boosting basée sur le gradient (*BoostGrad*), qui produit le taux d'erreur absolue moyenne (MAE) égale à 0.244.

3.6.3 Validation

Pour confirmer les performances de notre approche, nous avons réalisé une comparaison en variant le nombre de facteurs en entrée. Dans cette comparaison trois cas de figures sont testés comme indiqué dans le Tableau (3.5).

Cas	Symbole	Facteurs utilisés
Tous les facteurs	<i>Tous</i>	11 (Tous les facteurs du Tableau (3.4))
Choix aléatoire	<i>Aléatoire</i>	5 (f_{BR} , f_{Size} , f_{FL} , f_{AL} et f_{CT})
Approche proposée	<i>Proposition</i>	5 (f_{Resol} , f_{FR} , f_{FL} , f_{BR} et f_{AR})

TABLE 3.5 – Tableau comparatif de l'impact des facteurs.

La Figure (3.10) présente les résultats de comparaison, où le taux moyen d'erreur absolue est calculé pour la meilleure méthode d'apprentissage automatique utilisée (Section 3.4) dans les 3 cas présentés dans le Tableau (3.5).

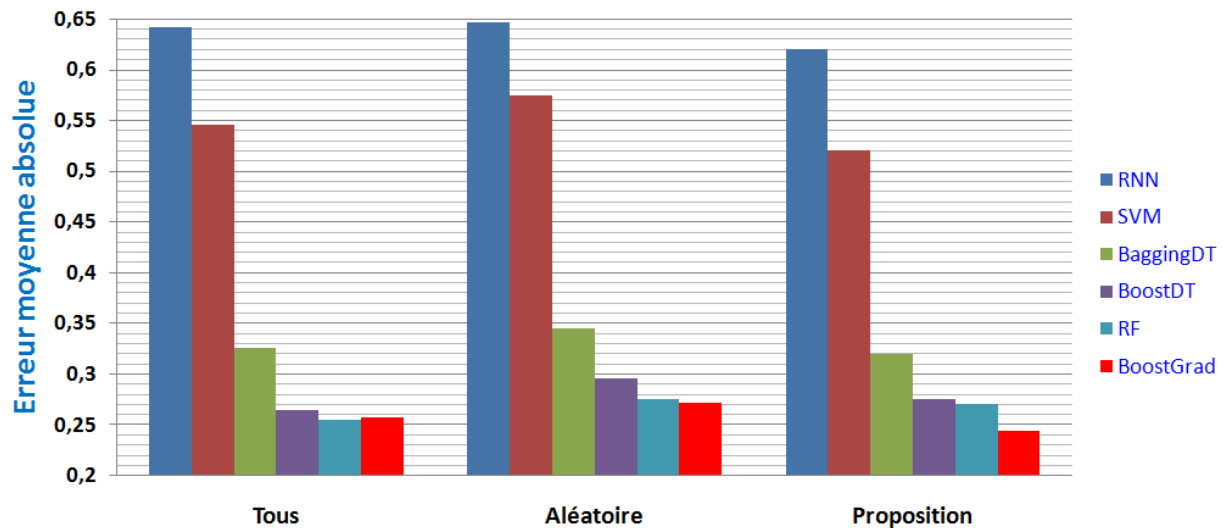


FIGURE 3.10 – Comparaison des approches de la sélection de facteurs

3.6.4 Discussion

Lors de l'évaluation des méthodes d'apprentissage automatique réalisée précédemment, nous avons observé que les méthodes basées sur le boosting et le bagging donnent de meilleures performances d'estimation de la QoE usager. Ceci est vérifié à la fois lors de l'utilisation de l'erreur absolue moyenne (MAE) (Figure (3.10)). Cela peut être justifié par le fait que les méthodes classiques d'apprentissage automatique utilisées (SVM et RNN) ont utilisé un nombre réduit de configurations (Tableau (3.2)) pendant les étapes d'apprentissage.

Par ailleurs, nous observons dans la Figure (3.10) que l'approche proposée (*Proposition*) pour sélectionner les facteurs donne de meilleurs résultats en termes d'erreur absolue moyenne (MAE) que les deux autres configurations testées (utilisation de tous les facteurs (*Tous*) et choix aléatoire des facteurs (*Aléatoire*)). Cela est vérifiée pour toutes les méthodes d'apprentissage automatique. Parmi celles-ci, la meilleure méthode résultante est le boosting basé sur le gradient

(BoostGrad)[61], qui utilise comme entrée les facteurs sélectionnés par notre approche de sélection (*Proposition*). Cette méthode donne le plus faible taux d'erreur égale de 0.224 comme nous l'avons présenté dans la partie précédente. Elle améliore le résultat obtenu par la meilleure méthode qui considère tous les facteurs de 4,5% (méthode forêts aléatoire utilisant tous les paramètres) et par rapport à la meilleure méthode utilisant cinq paramètres aléatoires (*Alatoire*) de 10%. En outre, notre approche permet d'avoir un taux de corrélation qui dépasse les 90%, ce qui peut être considéré comme un bon résultat pour l'estimation de la QoE usager.

Lors de l'exécution de notre approche de sélection de facteurs proposée dans la partie précédente, on notera que malgré que la méthode *BoostGrad* produit le meilleur taux d'erreur global, il a été observé que celle-ci ne trouve pas le meilleur MOS prédit (valeur la plus proche du MOS réel) pour certains échantillons de données comme il est illustré dans la Figure (3.11), qui représente la taux des MOS prédits le plus proche du MOS réel par chaque méthode d'apprentissage automatique.

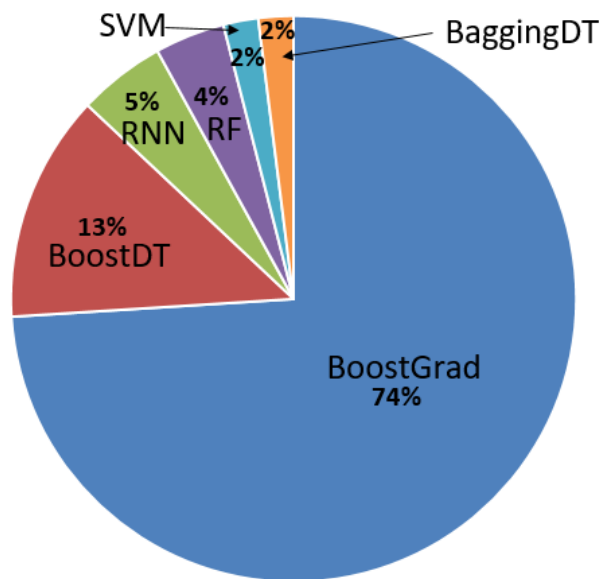


FIGURE 3.11 – Distribution du taux des MOS prédits le plus proche du MOS réel

D'après la Figure (3.11), nous observons que dans 74% des cas approximativement, la méthode de boosting basée sur le gradient ("BoostGrad") donne la valeur prédite du MOS la

plus proche de celle fournie par l'utilisateur. Cela confirme notre conclusion de la Figure (3.10), concernant la performance de cette méthode en termes de l'erreur absolue moyenne (MAE). Cependant, d'autres méthodes d'apprentissage automatique produisent de meilleurs résultats dans 26% (*SVM* (2%), *RNN* (5%), *RF* (4%), *BaggingDT* (2%) et *BoostDT* (13%)) des échantillons de la base de données utilisées.

Intuitivement, il peut donc être intéressant de combiner ces méthodes, en essayant de calculer chaque MOS prédit par la meilleure méthode en fonction du contexte afin d'améliorer l'estimation globale de la QoE usager. Ainsi, nous pouvons trouver la meilleure valeur prédite de la QoE de l'utilisateur en combinant toutes les valeurs prédites ou en choisissant la meilleure, comme nous le verrons dans le chapitre suivant.

3.7 Conclusion

Ce chapitre est consacré à l'utilisation des méthodes d'apprentissage automatique pour l'évaluation de l'expérience usager. Nous avons proposé, en premier lieu, notre première contribution. Elle consiste en une approche de sélection de facteurs les plus pertinents à utiliser par les méthodes d'apprentissage automatique pour calculer l'expérience usager. Cette approche permet d'optimiser le temps de calcul lors de l'estimation de l'expérience usager en utilisant un sous ensemble de facteurs au lieu d'utiliser tous les facteurs récoltés. En deuxième lieu, nous avons détaillé les méthodes d'apprentissage automatique utilisées dans cette thèse. Deux types de méthodes ont été étudiées, les méthodes dites classiques (les réseaux de neurones et les machines à vecteurs de support) et les méthodes dites ensemblistes (boosting et bagging). Pour finir, nous avons présenté une évaluation de ces méthodes ainsi que notre approche de sélection de facteurs en utilisant la base de données présentée dans le chapitre 2 et qui est récoltée dans l'environnement contrôlé. Les facteurs sélectionnés dans cette évaluation seront utilisés dans le chapitre suivant pour présenter notre deuxième contribution qui consiste en un nouveau modèle de calcul de la QoE usager.

Chapitre 4

Méthode d'estimation de la perception usager basée sur les fonctions de croyances

4.1 Introduction

Les études comparatives que nous avons menées confirment que les performances des méthodes d'apprentissage automatique pour l'estimation de la perception de l'utilisateur dépendent de plusieurs paramètres dont le contexte, le service ou encore les données d'apprentissage. Nous avons également constaté qu'une méthode performante dans un contexte peut ne pas l'être dans un autre. Intuitivement, il peut donc être intéressant de combiner plusieurs méthodes afin d'améliorer les estimations. L'objectif de ce chapitre est de présenter un nouveau modèle d'estimation de la perception usager (QoE). Ce modèle que nous appellerons CBF (*Combined Belief Functions*) combine l'utilisation de plusieurs méthodes issues de l'apprentissage automatique en utilisant la théorie des fonctions de croyance et la fusion d'informations.

4.2 Les fonctions de croyance

Les fonctions de croyance représentent une théorie mathématique basée sur la notion de preuves. Cette théorie a été développée en 1976 par Shafer [146] à la suite des travaux de Dempster sur les probabilités supérieures et inférieures. Shafer a en effet posé la théorie de base des fonctions de croyance pour la modélisation de la connaissance incertaine. Le but est de permettre de combiner des preuves distinctes pour calculer la probabilité d'un évènement. Toutefois, le cadre de cette théorie reste difficile à cerner à cause des relations complexes qui peuvent subsister entre les évènements [55]. Cela est probablement dû aux confusions qui peuvent découler des différences d'interprétation du terme "croyance". Plusieurs modèles de croyances co-existent dans la littérature, parmi elles, on peut citer :

- ▶ le modèle de Dempster [52].
- ▶ les limites supérieures et inférieures du modèle de probabilité [132].
- ▶ le modèle des Croyances Transférables (MCT) [131].

Plus de détails sur ces modèles de croyances peuvent être trouvés dans [55].

D'un point de vue mathématique, nous pouvons définir les fonctions de croyances comme suit :

Soit Θ un ensemble fini contenant toutes les valeurs possibles d'une variable X , appelé "*cadre de discernement*". Chaque structure de croyance sur Θ correspond à une "*fonction de masse*" notée " m " qui représente une distribution de probabilités généralisées sur les sous-ensembles de Θ ($2^\Theta \rightarrow [0, 1]$) que nous notons (A) , où :

$$\sum_{A \subset \Theta} m(A) = 1 \quad (4.1)$$

La fonction de masse (m) est définie pour chaque source d'information du système (appelée aussi oracle) et joue un rôle clé dans la construction du modèle. La différence entre le modèle MCT et les autres modèles (le modèle Dempster et le modèle des limites supérieures et inférieures du modèle de probabilité) réside dans le fait que les masses peuvent être affectées à chaque sous-ensemble $A \subset \Theta$ et non seulement aux singletons de Θ , comme c'est le cas dans la théorie des probabilités.

Dans notre cas, on s'intéresse à l'estimation du ressenti de l'utilisateur d'un service vidéo. Ce dernier est régi par un grand nombre de paramètres et par l'impact que peuvent avoir ces paramètres entre eux et sur l'utilisateur (Section 3.2). Le modèle MCT nous a donc paru le plus approprié pour estimer le ressenti utilisateur en combinant l'utilisation de plusieurs méthodes d'apprentissage automatique que nous utiliserons comme des "oracles". Dans ce qui suit nous présentons une synthèse pour la compréhension de ce modèle.

4.3 Modèle des Croyances Transférables

Le modèle de croyance transférable (MCT) appelé aussi "*TBM*" pour *Transferable Belief Model* a été développé par Philippe Smets en 1993 [131]. Ce modèle est un cadre formel pour la représentation et la combinaison des connaissances. Il est basé sur la définition des fonctions de croyances fournies par des sources d'information pouvant être élémentaires, redondantes ou éventuellement non-indépendantes. Il propose en outre un ensemble d'opérateurs permettant de combiner ces fonctions. Dans la pratique, le modèle MCT peut être employé dans le cadre de la fusion d'informations pour améliorer l'analyse et l'interprétation des données issues de sources d'informations multiples.

Contrairement aux modèles de croyance de Dempster [52] et le modèle à base de probabilités [132], le modèle MCT se distingue par son caractère fondamentalement non probabiliste où une justification axiomatique cohérente des principaux concepts de la théorie des fonctions de croyance est donnée. Cette justification a permis de clarifier le lien existant entre la représentation des croyances et la prise de décision [155]. Ainsi, les mécanismes de raisonnement du MCT sont donc regroupés en deux niveaux comme illustrés sur la Figure (4.1).

Le premier niveau appelé "Crédal" se porte, d'une part, sur la représentation des connaissances (partie statique), d'autre part, sur les combinaisons et le raisonnement sur ces connaissances (partie dynamique). Le deuxième niveau, appelé "Pignistique", concerne la prise de décision en prenant éventuellement en compte le risque et/ou le gain associés à cette décision.

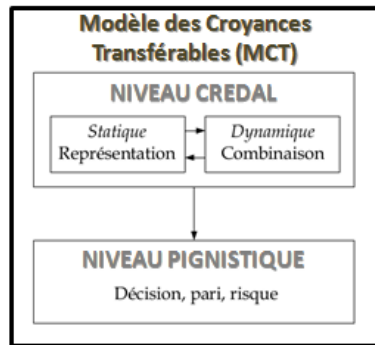


FIGURE 4.1 – Mécanismes du modèle des croyances transférables

1) Niveau crédal

Le niveau crédal représente la modélisation des croyances sous forme de fonctions de masse (m) et le niveau pignistique consiste à transformer ces fonctions en probabilités pour prendre des décisions. Selon [152], il n'existe pas de solution simple pour définir les fonctions de masse. Dans [21], les auteurs expliquent aussi qu'il n'existe pas de méthode générique pour la définir à moins qu'un expert ne donne son avis sous forme de croyance directement. De plus, la représentation des valeurs de croyance (" $m(A)$ ") est fortement dépendante de la modélisation de l'application, et la fonction de masse (" m ") est donc spécifique à cette application. Cependant, deux grandes familles émergent pour définir " m " indépendamment de l'application.

- La première famille utilise les mesures de dissimilarité qui se basent sur 2 types : (i) l'approche globale introduite par Shafer [146] et (ii) l'approche séparable proposée par Smets [154] et améliorée par Appriou [14].
- La deuxième famille initiée par Denœux [159], définit les fonctions " m " en utilisant des distances. Les méthodes de cette famille se concentrent sur les tendances basées sur les centroïdes, où les taux de croyance sont calculés par rapport au centroïde.

Nous notons aussi l'existence d'une autre approche basée sur une méthode heuristique dans laquelle le calcul de la fonction " m " est basé sur la valeur de croyance des singletons du cadre de discernement (Θ). La croyance est ensuite distribuée sur les sous-ensembles (A) qui composent le cadre Θ .

2) Niveau pignistique

Une fois la croyance modélisée sous forme de "m" pour chaque source d'information dans le niveau Crédal du modèle MCT, la prise de décision au niveau Pignistique commence. En effet, une décision doit être prise en utilisant plusieurs fonctions de masse m_i construites pour les différentes sources d'information. Une transformation ou une combinaison des $\{m_i\}$ est réalisée en fonction de la probabilité pour prendre ainsi la meilleure des décisions [131]. Dans ce niveau on distingue deux étapes :

- **2.1) Combinaison** : Elle consiste à synthétiser, sous la forme d'une fonction de croyance unique (nouvelle fonction de masse : m_{rule}) construite par fusion d'un ensemble de connaissances (à partir des : m_i) fourni par différents oracles tel que présenté dans l'équation (4.2).

$$m_{rule} = Combination(m_i) \quad (4.2)$$

Avec :

- i : l'index entre 1 et le numéro de la source d'information utilisée (oracle).
- **Combinaison** : une des règles de fusion d'information.

Une grande variété de règles de combinaison existent dans la littérature comme : la règle de Florea [59], la règle de Dempster-Shafer [55], la règle de Murphy [119] et la règle conjonctive de Smets [86].

- **2.2) Prise de décision** :

Dans cette étape, une décision doit être déduite de la connaissance (croyance) résultante (m_{rule}) pour obtenir la meilleure valeur. Habituellement, le maximum de la crédibilité (Bel), la plausibilité (pl) ou la probabilité pignistique ($betP$) sont pris en compte dans l'espace des décisions recevables. La fonction de crédibilité (Bel) est donnée à tous les éléments $X \in 2^\Theta (m_{rule})$ par :

$$Bel(X) = \sum_{Y \in 2^X, Y \neq \emptyset} m(Y) \quad (4.3)$$

- La fonction de la plausibilité (pl) est définie pour chaque $X \in 2^\Theta$ par :

$$pl(X) = \sum_{Y \in 2^X, Y \cap X \neq \emptyset} m(Y) = Bel(\Theta) - Bel(X^c) \quad (4.4)$$

avec : X^c est la complémentaire de X .

- La probabilité pignistique est donnée pour chaque $X \in 2^\Theta$, avec $X \neq \emptyset$ comme suit :

$$betP(X) = \sum_{Y \in 2^X, Y \neq \emptyset} \frac{|Y \cap X|}{|Y|} * \frac{m(Y)}{1 - m(\emptyset)} \quad (4.5)$$

En résumé, la meilleure décision est obtenue par le modèle MCT en utilisant l'un des critères de décision suivants (e.g : Bel , pl or $betP$) tel que présenté dans l'équation (4.6).

$$\begin{aligned} DECISION_{best} &= GetDecision(m_{rule}) \\ &= GetDecision(Combination(m_i)) \end{aligned} \quad (4.6)$$

Avec :

- **GetDecision** : Un des critères de prise de décision (e.g : Bel , pl or $betP$).
- m_i : Une fonction de masse construite pour la valeur prédite du i^{ieme} oracle i^{ieme} qui représente une des sources d'information du système comme présenté dans la Figure (4.2).

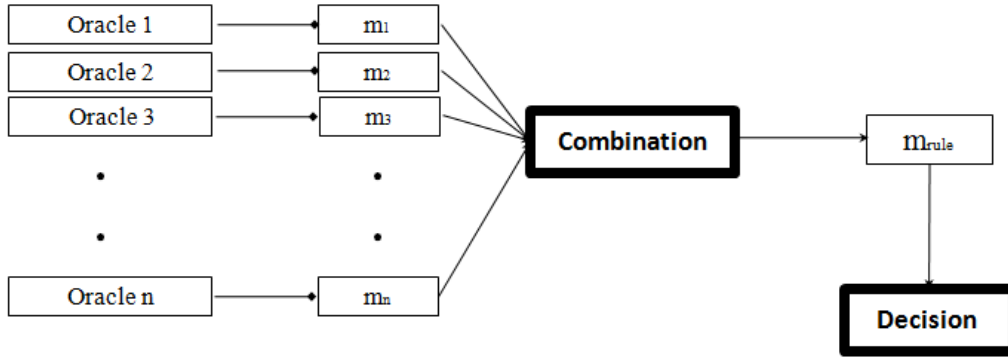


FIGURE 4.2 – Étapes du modèle des croyances transférables

4.4 Modèle proposé : Combined Belief Functions

Dans cette section, nous allons présenter notre contribution, nommée *CBF* (*Combination Belief Function*), qui consiste à proposer une nouvelle méthode d'estimation de l'expérience usager en utilisant plusieurs méthodes d'apprentissage automatique. Cette méthode se base sur le "modèle des croyances transférables (MCT)" [131]. Les sources d'informations, que nous appelons "oracles", sont les méthodes d'apprentissage automatique[26, 31]. Les informations à fusionner sont les résultats du calcul de la QoE en terme de MOS de chaque méthode. Dans cette section nous allons présenter le modèle CBF, qui est illustré dans la Figure (4.3) représentée ci-dessous, les notations utilisées sont dans le Tableau (4.1) :

Notations

-
- * Θ : Cadre de discernement.

 - * ML_i : Le i^{ieme} oracle utilisé (méthode d'apprentissage automatique).

 - * m_i : Fonction de masse du i^{ieme} oracle.

 - * t_k : Temps instantané.

 - * $MOS_{real}^{t_k}$: Le MOS réel donné par l'utilisateur à l'instant t_k .

 - * $MOS_{pred_i}^{t_k}$: La valeur calculée du MOS donnée par ML_i .

 - * $MOS_{best}^{t_k}$: La meilleure valeur calculée du score MOS ($MOS_{real}^{t_k}$).

TABLE 4.1 – Notations mathématiques

Le CBF utilise comme entrée les facteurs impactant la QoE, notés " $\{QoEIF_i\}$ " pour construire les différents oracles (ML_i). Chaque oracle calcule une valeur du MOS ($MOS_{pred_i}^{t_k}$). À partir de ces différentes valeurs ($MOS_{pred_i}^{t_k}$), une fonction de masse (m_i) est construite au niveau Crédal du modèle de croyance MCT . Dans le niveau Pignistique, un ensemble de règles de fusion est testé pour avoir la meilleure valeur de la QoE notée : $MOS_{best}^{t_k}$. La Figure (4.3) illustre les étapes importantes nécessaires à la mise en œuvre du modèle de prévision de CBF. Ces étapes sont décrites ci-dessous.

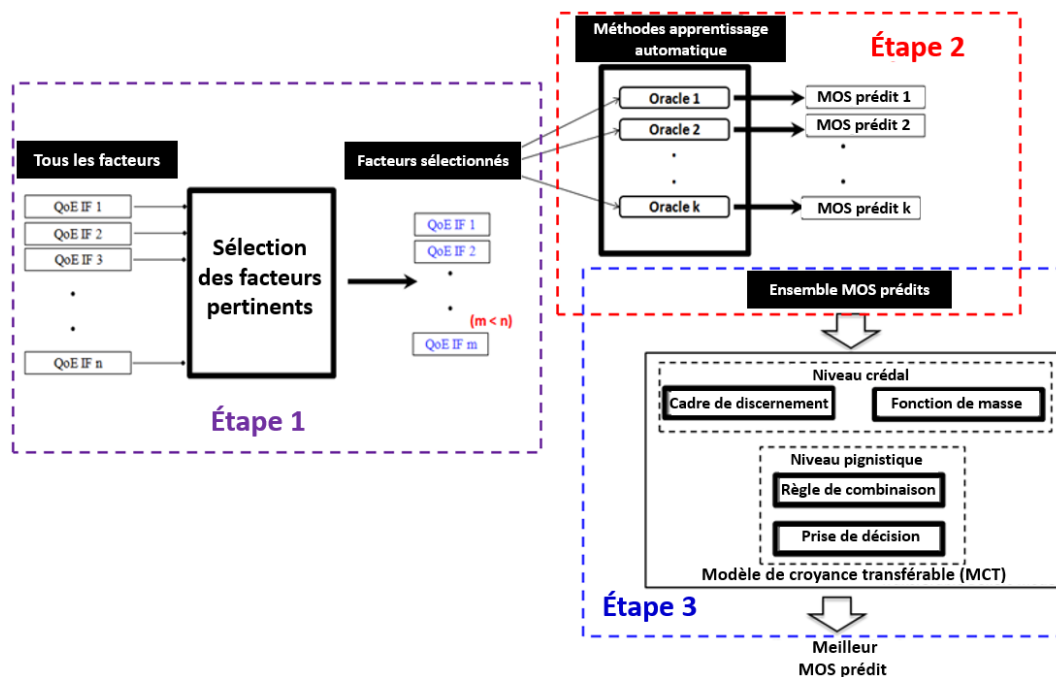


FIGURE 4.3 – Description des étapes du modèle CBF

► 01) Sélection des facteurs :

Cette première étape permet de sélectionner les facteurs les plus pertinents à utiliser comme entrée par les méthodes d'apprentissage automatique (ML_i). En outre, elle optimise le temps de calcul lors de l'estimation de l'expérience usager en utilisant un sous-ensemble de facteurs au lieu d'utiliser tous les facteurs récoltés. Dans cette étape, plusieurs types de méthodes de sélection peuvent être utilisées comme les méthodes de filtrage, enveloppantes (*wrapper*) ou intégrées (*embedded*) [49].

► **02) Construction des oracles :**

Dans cette deuxième étape, un ensemble de méthodes d'apprentissage automatique sont considérées comme oracles (sources d'information). Chaque méthode donne alors une valeur ($MOS_{pred_i}^{tk}$) en prenant comme entrée les facteurs sélectionnés lors de la précédente étape et en optant pour la meilleure configuration possible.

► **03) Implémentation du modèle des Croyances Transférables (MCT) :**

Une fois toutes les valeurs $MOS_{pred_i}^{tk}$ connues, nous les utilisons pour construire le modèle MCT [131]. Dans cette étape, les éléments qui composent le modèle MCT sont les suivants :

- Un cadre de discernement (Θ).
- Des fonctions de masses (m) sont définies en utilisant les valeurs calculées ($MOS_{pred_i}^{tk}$),
- Une ou plusieurs règles de fusion d'information sont testée(s) en utilisant les fonctions de masse (m_i) calculées précédemment. Le but de cette fusion est de synthétiser, sous la forme d'une fonction de croyance globale et unique (nouvelle fonction de masse : m_{rule}), l'ensemble des connaissances à partir des fonctions de masses calculées (m_i).
- Une décision devrait ensuite être prise pour obtenir la meilleure valeur prédite (MOS_{best}^{tk}) en utilisant l'un des critères de décision (e.g : *Bel*, *pl* or *betP*).

En résumé, la meilleure valeur prédite (MOS_{best}^{tk}) est obtenue en utilisant l'un des critères de prise de décision (exemple : maximum de crédibilité ou plausibilité) sur la fonction de masse globale (m_{rule}) construite à partir de la fusion des différentes masses " m_i ". L'équation (4.7) résume le modèle d'estimation proposé.

$$\begin{aligned}
 \text{Étape 1 : } & \{ x \in SelectedQoEIFs \subset QoEIFs \\
 \text{Étape 2 : } & \{ MOS_{pred_i}^{tk} = ML_i(SelectedQoEIFs) \\
 \text{Étape 3 : } & \left\{ \begin{array}{l} m_i = GetMassFun(MOS_{pred_i}^{tk}) \\ m_{rule} = Combination(m_i) \\ MOS_{best}^{tk} = GetDecision(m_{rule}) \end{array} \right. \quad (4.7)
 \end{aligned}$$

Avec :

- ML_i : Une des méthodes d'apprentissage automatique utilisée comme oracle (Section (3.3)).

- m_i : Fonction de masse définie pour la valeur prédite $MOS_{pred_i}^{tk}$ calculée par ML_i .
- m_{rule} : Fonction de masse globale résultante de la combinaison de toutes les fonctions de masse m_i .
- **GetMassFun** : Fonction de masse (m_i) définie pour quantifier les croyances du modèle MCT.
- **Combination** : Une des règles de fusion d'information (m_i) est utilisée pour construire la fonction de masse globale (m_{rule}).
- **GetDecision** : Un des critères de prise de décision (Bel , pl or $betP$).

4.5 Implémentation

4.5.1 Étapes

Le modèle CBF utilise les paramètres applicatifs issus du service vidéo (catégorie QoA (Figure 1.2)) comme entrée des oracles afin d'estimer le MOS des usagers (QoE). Les données utilisées sont issues de la base de données [93] construite dans un environnement contrôlé présenté dans le Chapitre 2. Dans cette partie, nous expliquons l'implémentation des différentes étapes du modèle CBF illustré par la Figure (4.4) ci-dessous.

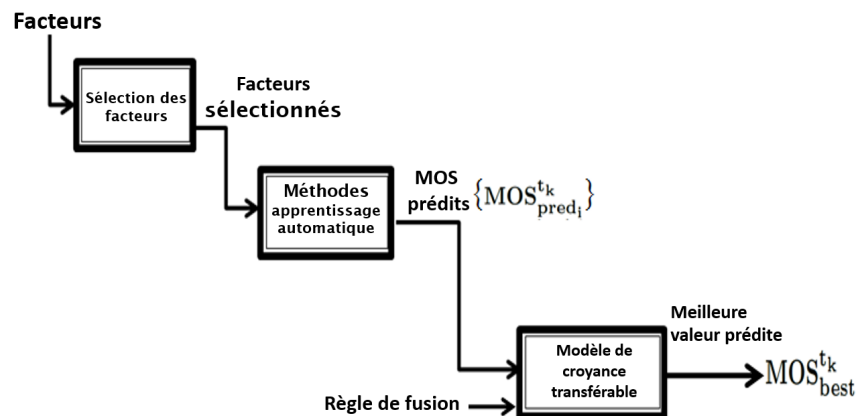


FIGURE 4.4 – Fonctionnement du modèle CBF

1) Sélection des facteurs :

La première étape concerne la sélection des facteurs à utiliser comme entrée par les méthodes d'apprentissage automatique. Dans cette étape, nous avons utilisé l'approche développée dans le Chapitre (3) et qui utilise une méthode de type enveloppante (*wrapper*)[49] pour sélectionner les caractéristiques. Cette approche se base sur la corrélation de Pearson[73].

À la fin du processus de sélection, cinq facteurs sont sélectionnés. Ces facteurs sont : la résolution vidéo (f_{Resol}), la fréquence d'images vidéo par seconde (f_{FR}), le taux de perte d'images par seconde (f_{FL}), le taux de débit binaire vidéo (f_{BR}) et le taux débit binaire audio (f_{AR}).

2) Construction des oracles :

Les oracles dans cette étape représentent les méthodes d'apprentissage automatique présentées dans le Chapitre (3). Ces méthodes sont les machines à vecteurs de support (SVM)[113], les réseaux de neurones (RNN)[32], les forêts aléatoires (RF)[37], le bagging basé sur les arbres de décisions (BaggingDT) [36], le boosting basé sur les arbres de décision (BoostDT) [25] et le boosting basé sur le gradient (BoostGrad)[61].

Dans cette deuxième étape, nous utilisons ces 6 oracles afin de calculer différentes valeurs de MOS prédits. À la fin de cette étape, chaque oracle donne une valeur prédite du MOS usager ($MOS_{pred_i}^{tk}$), qui sera utilisée dans la prochaine étape du modèle CBF.

3) Implémentation du modèle des Croyances Transférables :

Une fois que toutes les valeurs du MOS ($MOS_{pred_i}^{tk}$) sont calculées pour chaque oracle, nous les utilisons pour construire le modèle des Croyances Transférables (MCT) [131] qui forme la troisième étape de notre modèle CBF. Le modèle MCT est implémenté et les éléments qui le composent sont définis ainsi :

- Cadre de discernement représente les valeurs du score du MOS où : $\Theta = MOS = \{1, 2, 3, 4, 5\}$.
- En utilisant les valeurs prédites ($MOS_{pred_i}^{tk}$), la fonction de masse sera définie pour chaque valeur prédite. Dans notre cas, une approche heuristique est utilisée pour définir la fonction de masse "m" (voir la Section 4.5.2).
- Plusieurs règles de fusion d'information seront testées. Dans le cadre de nos travaux, les règles testées sont : Yager, Smets conjonctive, Dempster-Shafer's, Florea et Murphy pondérée [55, 86, 59].

- Pour la prise de décision, le maximum de la crédibilité (équation 4.5) est utilisé au niveau pignistique du modèle MCT.

4.5.2 Définition de la fonction de masse

Pour définir la fonction de masse utilisée dans le modèle de croyance MCT, nous avons proposé une heuristique comme indiqué dans la Section (4.4). L'idée consiste à déterminer la valeur de croyance pour chaque singleton du cadre de discernement (Θ), et à distribuer ces valeurs sur le reste des sous-ensembles de croyances de Θ comme expliqué ci-dessous.

- A) Les singletons du cadre de discernement Θ constituent les valeurs du score MOS.

$$\Theta = MOS = \{1, 2, 3, 4, 5\}.$$

- B) Pour chaque oracle, on calcule les valeurs MOS prédites ($MOS_{pred_i}^{tk}$). Sachant que les scores MOS sont des entiers entre 1 et 5, chaque valeur $MOS_{pred_i}^{tk}$ est comprise entre un entier inférieur et un entier supérieur comme suit :

$$MOS_{inf} \leq MOS_{pred_i}^{tk} \leq MOS_{sup}, \text{ avec } MOS_{inf} = MOS_{sup} - 1$$

- C) Pour chaque singleton de Θ , la fonction de masse (m_i) pour chaque valeur prédite ($MOS_{pred_i}^{tk}$) est définie avec l'équation (4.8) :

$$m : \begin{cases} m_{MOS_{sup}} = 1 - (| MOS_{pred_i}^{tk} - MOS_{sup} |) \\ m_{MOS_{inf}} = 1 - (| MOS_{pred_i}^{tk} - MOS_{inf} |) \\ m_v = 0 // o : v \in (\Theta / \{MOS_{inf}, MOS_{sup}\}) \end{cases} \quad (4.8)$$

Exemple :

- Posant l'oracle (ML_i) comme la méthode d'apprentissage automatique "Forêts aléatoires (RF)". Supposant qu'elle donne une valeur calculée du score MOS égale à : $MOS_{pred_i}^{tk} = 3, 24$. Alors, on déduit que $MOS_{sup} = 4$ et $MOS_{inf} = 3$. Par conséquent, la fonction de masse pour

les singletons (au début) est :

$$m : \begin{cases} m_1 = 0 \\ m_2 = 0 \\ m_3 = 1 - (| \text{MOS}_{\text{pred}_i}^{\text{tk}} - \text{MOS}_{\text{inf}} |) = 1 - (| 3.24 - 3 |) = 0.76 \\ m_4 = 1 - (| \text{MOS}_{\text{pred}_i}^{\text{tk}} - \text{MOS}_{\text{sup}} |) = 1 - (| 3.24 - 4 |) = 0.24 \\ m_5 = 0 \end{cases} \quad (4.9)$$

D) Pour répartir la valeur de croyance du modèle MCT sur les sous-ensembles de Θ , nous considérons simplement les deux voisins singletons qui peuvent avoir une partie de la valeur de croyance. Cela peut être justifié par le fait que, $\text{MOS}_{\text{inf}} \leq \text{MOS}_{\text{pred}_i}^{\text{tk}} \leq \text{MOS}_{\text{sup}}$, comme illustré dans la Figure (4.5).

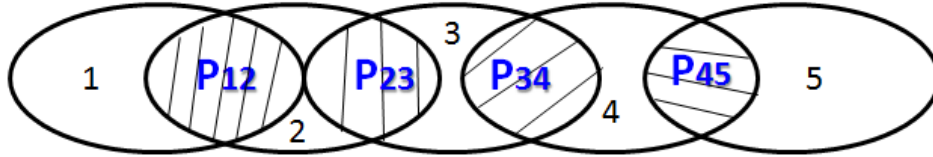


FIGURE 4.5 – Distribution des masses du modèle CBF

Dans un processus simple, nous considérons non seulement les 5 valeurs entières du MOS, mais nous prenons aussi en compte toutes les unions possibles de deux classes voisines qui peuvent avoir une valeur de croyance. Dans la pratique, toutes les valeurs de masse m_6 à m_{15} (2 des classes voisines) sont calculées comme suit :

$$m : \begin{cases} m_6 = m_1 * m_2 \\ m_{10} = m_2 * m_3 \\ m_{13} = m_3 * m_4 \\ m_{15} = m_4 * m_5 \\ m_j = 0 // j = 7 : 31 \text{ et } (j \neq 10) \text{ et } (j \neq 13) \text{ et } (j \neq 15) \\ m_{32} = m_0 = 0 \end{cases} \quad (4.10)$$

Exemple :

Sachant que la valeur du score MOS calculé (3.24) par la méthode RF est entre les deux scores 3 et 4, donc c'est le m_{13} qu'il faut calculé. Pour ceci, nous utilisons les valeurs de $m_3 = 0.76$ et de $m_4 = 0.24$ issues de l'étape précédente comme suit :

$$m : \begin{cases} m_{13} = m_3 * m_4 = 0.182 \\ m_j = 0 // (j = 6 : 32) \text{ et } (j \neq 13) \end{cases} \quad (4.11)$$

E) Enfin, la valeur de croyance du singleton sera mise à jour en raison de la distribution des croyances, comme indiqué ci-dessous dans l'équation 4.12) :

$$m : \begin{cases} m_1 = m_1 - m_6 \\ m_2 = m_2 - \frac{1}{2}((m_6 + m_{10}) + (m_6 * m_{10})) \\ m_3 = m_3 - \frac{1}{2}((m_{10} + m_{13}) + (m_{10} * m_{13})) \\ m_4 = m_4 - \frac{1}{2}((m_{13} + m_{15}) + (m_{13} * m_{15})) \\ m_5 = m_5 - m_{15} \end{cases} \quad (4.12)$$

Justification :

- m_1 : La valeur de croyance pour avoir un $MOS = 1$ est égale à la valeur initiale m_1 moins la valeur de croyance pour avoir un $MOS = 1$ et un $MOS = 2$ en même temps.
- m_3 : La valeur de la croyance pour avoir un $MOS = 3$ est égale à la valeur initiale m_3 moins la moitié de la somme des deux valeurs de croyance : (i) la valeur d'avoir un $MOS = 2$ et un $MOS = 3$ dans le même temps et (ii) la valeur d'avoir un $MOS = 3$ et un $MOS = 4$ dans le même temps.

Exemple :

Pour mettre à jour les valeurs de croyance des singletons (m_1 à m_5) de Θ , nous allons distribué les valeurs des croyances des masses allant de m_6 à m_{32} sur les masses des singletons (m_1 à m_5). D'après l'équation ((4.11), nous observons que toutes les masses à l'exception de m_{13} sont nulles. Ainsi, les valeurs m_3 et m_4 seront mises à jour comme suit dans l'équation (4.11) :

$$m : \begin{cases} m_1 = m_1 - m_6 = 0 \\ m_2 = m_2 - \frac{1}{2}((m_6 + m_{10}) + (m_6 * m_{10})) = 0 \\ m_3 = m_3 - \frac{1}{2}((m_{10} + m_{13}) + (m_{10} * m_{13})) = 0.76 - \frac{1}{2}(0 + 0.18) + 0 = 0.67 \\ m_4 = m_4 - \frac{1}{2}((m_{13} + m_{15}) + (m_{13} * m_{15})) = 0.24 - \frac{1}{2}(0 + 0.18) + 0 = 0.15 \\ m_5 = m_5 - m_{15} = 0 \\ m_{13} = 0.18 \\ m_j = 0 // (j = 6 : 32) \text{ et } (j \neq 13) \end{cases} \quad (4.13)$$

Enfin, nous aurons les valeurs de croyances ci-dessous :

m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}	m_{11}	m_{12}	m_{13}	m_{14}	m_{15}	m_{16}	m_{17}	m_{18}	m_{19}	m_{20}	...	m_{32}	Σ	
0	0	0.67	0.15	0	0	0	0	0	0	0	0	0.18	0	0	0	0	0	0	0	0	...	0	1

TABLE 4.2 – Taux des croyances résultant pour une valeur calculée du score MOS

4.5.3 Mise en œuvre de la règle pondérée de Murphy

Dans la troisième étape du modèle CBF, et plus exactement au niveau pignistique du modèle de croyance MCT, cinq règles de fusion d'information sont testées. Ces règles (Yager, conjonctive de Smets, Dempster-Shafer, Florea et Murphy pondérée) sont présentées et détaillées dans les études [86, 59, 119]. Ces méthodes sont souvent utilisées pour des problématiques semblables à la notre[55].

Concernant la règle pondérée de Murphy, Murphy et al. proposent dans [119] une règle de combinaison initiale comme la moyenne des affectations de croyance de base de tous les éléments X de Θ ($X \in 2^\Theta$) définies par :

$$m_{Mean} = \frac{1}{M} \sum^M m_i(X) \quad (4.14)$$

Dans le modèle de calcul CBF, cette règle est modifiée afin d'améliorer l'estimation du ressenti usager (QoE) en pondérant chaque oracle individuellement selon sa performance de prédiction (α_i) :

1) En se basant sur la performance de prédiction des oracles ML_i (taux d'erreur global de chaque oracle (MAE_i)), on calcule un poids pour chaque oracle ML_i .

2) En utilisant le poids calculé, la règle de base de Murphy est modifiée comme suit :

$$m_{Mean} = \frac{1}{M} \sum^M \alpha_i * m_i(X) \quad (4.15)$$

Avec α_i qui dépend du taux d'erreur global MAE_i de chaque oracle ML_i et $\sum(\alpha_i) = 1$.

4.6 Évaluation

Les données recueillies au cours de la campagne d'évaluation de la QoE en environnement contrôlé (Section (2.3)) sont utilisées pour évaluer les performances du modèle proposé. Ce dernier utilise 6 méthodes d'apprentissage automatique comme oracles et cinq règles de combinaisons. Les oracles utilisés sont les réseaux neuronaux aléatoires (RNN)[32], les machines à vecteurs de support (SVM)[113], les méthodes de boosting (le boosting basé sur les arbres de décision (BoostDT) [25] et le boosting basé sur le gradient (BoostGrad)[61]) et les méthodes de bagging (les forêts aléatoires (RF)[37], le bagging basé sur les arbres de décisions (BaggingDT) [36]) (voir Section (3.3)). Les règles de combinaison testées sont la règle de Florea [59], la règle de Dempster-Shafer [55], la règle conjonctive de Smets [86] et la règle de Yager[86]. Les facteurs sélectionnés ($SelectedQoEIFs$) comme entrées pour les différents oracles sont : la résolution vidéo (f_{Resol}), le taux d'images vidéo (f_{FR}), le taux de perte d'images vidéo (f_{FL}), le débit binaire vidéo (f_{BR}), et le débit binaire audio (f_{AR}). À l'aide de ces $SelectedQoEIFs$, nous calculons les valeurs de la QoE en termes de scores MOS et nous analysons les performances d'estimation (entre MOS calculé et MOS réel).

Nous considérons deux cas pour l'évaluation. Dans le premier cas, on augmente progressivement la quantité de données d'apprentissage (10% à chaque étape). Dans le deuxième cas, on utilise la totalité de la base de données [93]. Dans les deux cas testés, nous avons utilisé comme indicateur de performance l'erreur absolue moyenne (MAE) comme le suggère [6, 158], qui

explique que cet indicateur peut être un meilleur indicateur que l'erreur quadratique (*RMSE*) dans certains cas.

Dans le premier cas, les résultats des calculs en termes de taux d'erreur absolue moyenne (MAE) pour chaque oracle et pour chaque règle utilisée dans le modèle CBF sont présentés dans la Figure (4.6). L'axe des abscisses représente le % utilisé dans l'étape d'entraînement, tandis que l'axe des ordonnées représente le taux d'erreur absolue moyenne (MAE) pour chaque méthode/règle présentée avec plusieurs couleurs.

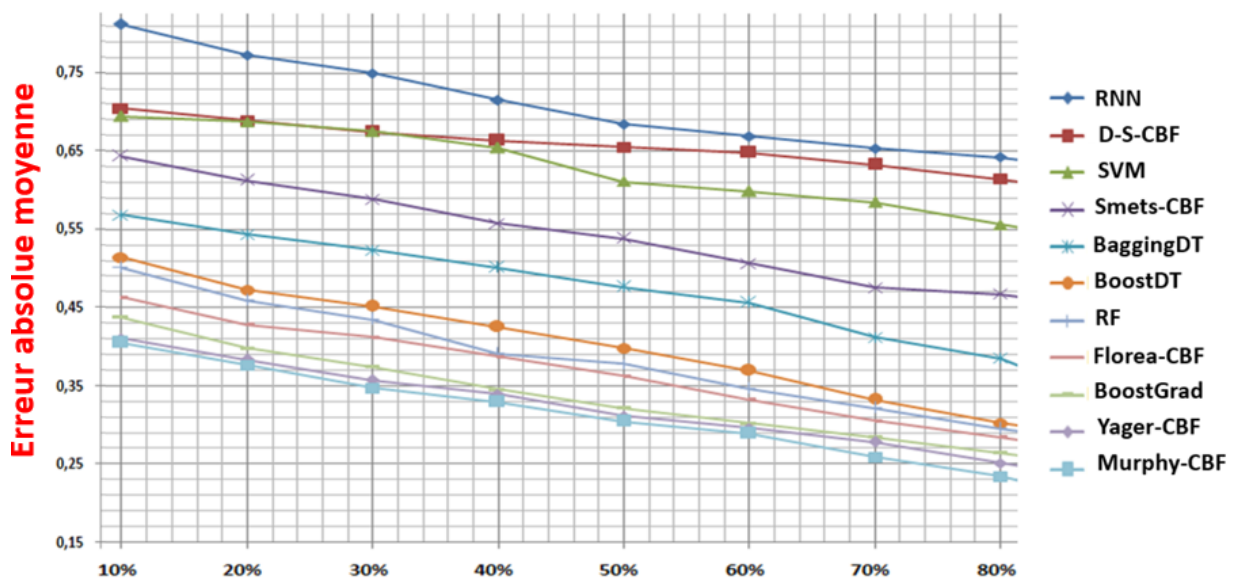


FIGURE 4.6 – Performances en utilisant une taille variable de la base de données

On constate, dans ce premier cas, qu'à l'exception des règles de Smets (règle conjonctive de Smets) et de Dempster-Shafer (règle D-S) (règles de fusion de base), l'utilisation des règles de combinaison produit de bons résultats pour calculer les scores MOS. Dans cette figure, nous avons également observé qu'en utilisant 10% de la base de données globale, la meilleure méthode d'estimation est le modèle CBF utilisant la règle de Murphy pondérée avec un taux d'erreur absolue moyenne de 0,39 et une amélioration de 13,5% par rapport à la méthode "BoostGrad" (meilleure méthode d'apprentissage automatique) et de 14% par rapport à RF

(meilleure méthode basée sur le bagging). La deuxième meilleure méthode est le modèle CBF utilisant la règle Yager.

En augmentant la taille de la base de données au fur et à mesure de 10% à chaque étape, on observe que le modèle CBF réduit le taux d'erreur absolue moyenne (MAE). Ces résultats sont justifiés par l'amélioration de la prédiction de chaque oracle en fonction de la quantité des données utilisée pendant la phase d'apprentissage.

À la fin, nous utilisons la totalité de la base de données pour apprendre le meilleur oracle (*BoostGrad*) avec le modèle CBF avec les 5 règles de combinaison. La Figure (4.7) présente les résultats obtenus en termes de taux d'erreur absolue moyenne (MAE).

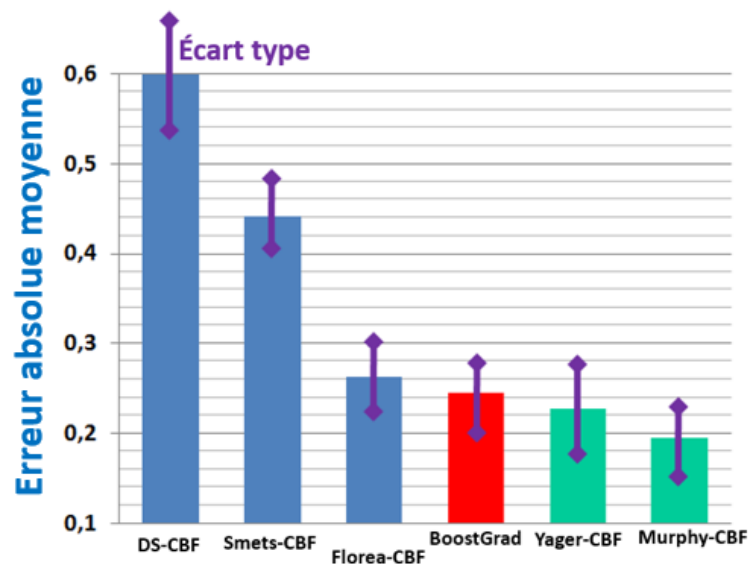


FIGURE 4.7 – Performances en utilisant la base de données complète

Dans cette évaluation, nous avons considéré 90% de la base de données collectée pour la phase d'apprentissage et 10% pour la validation. Dans ce cas, nous observons que les règles des combinaisons Dempsfer Shafer ("DS-CBF"), Smets ("Smets-CBF") et Florea ("Florea-CBF") ont respectivement 0,59, 0,44 et 0,26 de taux d'erreur absolue moyenne (MAE), ce qui donne de moins bons résultats que l'oracle "BoostGrad". Ces règles de base ne sont pas adaptées à

l'évaluation de la QoE dans pour notre jeu de données. En outre, nous remarquons que les règles de Murphy pondérée ("Murphy-CBF") et de Yager ("Yager-CBF") se comportent mieux que toutes les méthodes étudiées. Dans le cas de la règle Yager ("Yager-CBF"), la prédiction est améliorée de 7% par rapport à l'oracle "BoostGrad". Quant à la règle de combinaison Murphy pondérée ("Murphy-CBF"), l'erreur absolue moyenne est inférieure à 0,2, ce qui se traduit par une amélioration de 20,5% par rapport à l'oracle "BoostGrad".

4.6.1 Discussion

L'utilisation des règles de combinaison pour fusionner les résultats de plusieurs oracles nous permet d'obtenir de bien meilleurs résultats comme on peut le voir dans le Tableau (4.3). Le taux de corrélation "r" est calculé à l'aide de l'équation suivante (4.16) :

$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[n \sum x^2 - (\sum x)^2][n \sum y^2 - (\sum y)^2]}} \quad (4.16)$$

Avec :

- $y \in \{\text{MOS}_{\text{real}}^{t_k}\}$ qui représente le score MOS réel,
- $x \in \{\text{MOS}_{\text{pred}_i}^{t_k}\}$ qui représente le score MOS prédit par l'oracle 'i',
- 'n' le nombre d'exemples.

Dans le Tableau (4.3), nous présentons le coefficient de corrélation (équation (4.16)) et l'erreur absolue moyenne (MAE) pour étudier la qualité du modèle. En outre, nous y ajoutons aussi le coefficient de détermination (r^2) pour confirmer les résultats et la qualité des différents modèles testés. Par ailleurs, les temps de calcul sont indiqués pour discuter des limites pratiques des approches testées. Ces valeurs sont calculées sur la base d'un ordinateur Dell (Intel R Core TM, i7 CPU avec 3.40 GHz et 8 Giga-octes de RAM testés sous Linux (Ubuntu 14.04)).

En pratique, le modèle linéaire nécessite le minimum de temps de calcul (< 0.001 millise-

Modèle	Forme du modèle	Qualité du modèle			Temps de calcul
		r	r^2	MAE	
LM	Régression linéaire MOS = 0.001071* resolution + 0.000247* video bit rate + 0.008184*frame rate - 0.023250*frame loss + 0.041053 * audio bit rate + 0.374665	0.80	0.652	0.693	< 0.001 (ms)
RNN	Apprentissage artificiel (oracle)	0.831	0.690	0.623	<0.003 (ms)
SVM	Apprentissage artificiel (oracle)	0.842	0.708	0.528	< 0.002 (ms)
RF	Apprentissage artificiel (oracle)	0.894	0.801	0.274	< 0.002 (ms)
BaggingDT	Apprentissage artificiel (oracle)	0.872	0.760	0.321	< 0.004 (ms)
BoostDT	Apprentissage artificiel (oracle)	0.887	0.786	0.279	< 0.004 (ms)
BoostGrad	Apprentissage artificiel (oracle)	0.904	0.826	0.244	< 0.010 (ms)
D-S-CBF	Théorie des fonctions et apprentissage artificiel	0.842	0.679	0.594	<0.025 (ms)
Smets-CBF	Théorie des fonctions et apprentissage artificiel	0.866	0.750	0.446	< 0.025 (ms)
Florea-CBF	Théorie des fonctions et apprentissage artificiel	0.895	0.801	0.260	< 0.025 (ms)
Yager-CBF	Théorie des fonctions et apprentissage artificiel	0.917	0.841	0.225	< 0.025 (ms)
Murphy-CBF	Théorie des fonctions et apprentissage artificiel	0.920	0.846	0.194	< 0.025 (ms)

TABLE 4.3 – Tableau comparatif des performances des méthodes utilisées

conde) en raison de sa formule simple qui ne nécessite pas de calcul complexe. Les méthodes SVM, RNN et RF ont des temps d'exécution inférieurs à 0,003 milliseconde en moyenne pour calculer la valeur du MOS. Concernant les méthodes de bagging et de boosting (à part RF), ces méthodes prennent plus de temps à calculer les données comme on peut le voir dans le tableau (4.3). En moyenne, elles nécessitent entre 0,004 et 0,010 millisecondes pour le calcul de la Qoe. Enfin, le temps de calcul de notre modèle CBF dépend du temps des calculs des différentes méthodes d'apprentissage automatique utilisées comme oracles et du temps de la fusion des valeurs. Dans la pratique, le temps de calcul total est obtenu en additionnant le temps que chaque oracle calcule sa valeur, avec le temps que prend la fusion des résultats (moins de <

0,002 millisecondes). Notez que le temps de combinaison dépend de la règle de fusion utilisée et la règle de Murphy pondérée est la meilleure et ne dépasse pas 0,001 milliseconde. Enfin, nous soulignons que le temps d'exécution de notre modèle de calcul de la QoE est inférieur à 0,025 millisecondes.

4.7 Conclusion

Dans ce chapitre, nous avons présenté un nouveau modèle de calcul de la QoE usager, en termes de scores MOS, appelé CBF (Combined Belief Functions). Le modèle proposé de prédiction se base sur la combinaison de plusieurs méthodes d'apprentissage automatique et l'utilisation des fonctions de croyance et la fusion d'informations. L'idée de notre proposition est d'utiliser, non pas une seule méthode pour estimer l'expérience usager, mais d'utiliser plusieurs méthodes d'apprentissage automatique et de combiner leurs résultats afin d'affiner les calculs de la QoE.

Dans notre étude, nous avons évalué notre proposition avec 6 méthodes d'apprentissage automatique (2 méthodes classiques, 4 méthodes ensemblistes). Les résultats des tests confirment que la combinaison de ces méthodes améliore la performance par rapport aux méthodes d'apprentissage automatique, qu'elles soient classiques (réseau de neurones, machines à vecteurs de support) ou ensemblistes (forêts aléatoires, boosting avec gardient, etc.). Dans le chapitre suivant, nous allons mettre en application le modèle CBF afin de proposer une solution de diffusion vidéo à débit variable (ABR).

Chapitre 5

Méthode pour la diffusion vidéo à débit variable

5.1 Introduction

L'objectif de ce chapitre est de présenter une nouvelle solution de diffusion vidéo à débit variable (Adaptive Bit Rate (ABR)), régulé par la QoE. Le calcul de ladite QoE se base sur la méthode proposée dans le chapitre précédent. Cette solution a pour but d'améliorer la qualité perçue par les utilisateurs et d'optimiser les ressources des opérateurs réseau. Pour ce faire, la solution se base sur un modèle paramétrique qui optimise en temps réel la qualité de la vidéo en fonction de la mesure instantanée de la QoE estimée.

Dans ce chapitre, nous commençons par introduire le service de diffusion vidéo à débit variable (ABR). Ensuite, nous présentons la phase de collecte des données subjectives utilisées pour l'estimation de la QoE usager. Cette estimation est ainsi utilisée pour concevoir une nouvelle solution de streaming adaptatif, notée "Q2ABR" (*Quality of Experience aware Adaptive Bit Rate*), qui se base sur l'apprentissage par renforcement (AR) dont le but est de sélectionner la représentation vidéo des prochains segments en fonction de la variabilité de la QoE. Enfin, les performances de celle-ci sont évaluées et comparées à des méthodes ABRs issues de l'état de l'art. Cette évaluation est faite sur la base de traces réelles.

5.2 Diffusion vidéo à débit variable

5.2.1 Stratégies d'adaptation de la diffusion vidéo

D'après le rapport 2017 de CISCO [50], le trafic vidéo représente plus de 75% du trafic mondial. Ce trafic est majoritairement dû aux services de la vidéo à la demande fournis par des plateformes telles que YouTube¹, DailyMotion² et Netflix³. Ce service servira à valider notre méthode de mesure de la QoE détaillée dans le Chapitre (4). Le fonctionnement de ce type de service est comme suit : l'utilisateur final effectue une requête HTTP via Internet, pour demander une vidéo particulière chez le fournisseur de service (YouTube, Netflix, etc.). Ce dernier sélectionne la vidéo ayant la qualité adéquate par rapport aux conditions réseau (changement de débit, variation de latence, etc.) et la lui transmet via l'infrastructure de l'opérateur réseau, qui joue le rôle du transitaire. Pour assurer un bon fonctionnement du service indépendamment des changements des conditions réseau, des stratégies d'adaptations (*Http Adaptive Streaming (HAS)*) sont utilisées. Ces stratégies peuvent être classées en trois catégories principales :

- **Stratégies basées sur le transcodage évolutif** : Dans cette catégorie, la compression et la résolution de la vidéo sont adaptées au moyen d'un transcodage instantané du contenu brut. Le but est de varier en temps réel le débit en fonction de la fréquence d'images. Cependant, cette méthode est difficile à mettre œuvre à grande échelle en raison des ressources qu'elle nécessite. [30].
- **Stratégies basées sur l'encodage évolutif** : Dans cette catégorie, les méthodes utilisées exploitent l'évolutivité spatiale et temporelle pour adapter la résolution de l'image. Pour cela, elles emploient des codecs évolutifs tels que H264/SVC afin de réduire la charge de traitement et améliorer les performances des stratégies d'adaptations (HAS). Cependant, ces méthodes nécessitent des serveurs spécialisés pour analyser en temps réel les contenus vidéo en vue d'adapter l'encodage en fonction des caractéristiques spatiales et temporelles des vidéos[122].

1. <https://www.youtube.com/>

2. <https://www.dailymotion.com/>

3. <https://www.netflix.com/>

- **Stratégies basées sur le changement de qualité** : Les méthodes de cette catégorie sont appelées également méthodes de diffusion de vidéos à débit variable (*Adaptive Bit Rate (ABR)*). Dans ces méthodes, la vidéo originale est découpée en segments de quelques secondes. Chaque segment est encodé en différents niveaux prédéfinis de qualités (appelé "*profils*") [185]. Ces différents segments sont stockés dans un serveur web. Ce serveur a un fichier "*manifest*" qui contient toutes les informations des représentations des segments vidéos (taille segment vidéo, profil utilisateur, etc.). Ce fichier est envoyé en premier lors de la connexion cliente. Après avoir téléchargé le fichier de "*manifest*", le client (lecteur vidéo) sélectionne à la volée le profil le plus adéquat en utilisant des requêtes GET-HTTP.

Le Tableau (5.1) représente un récapitulatif des 3 catégories des stratégies d'adaptations de vidéo. Nos travaux se sont focalisés sur la dernière catégorie. Ce choix est justifié d'une part, par la popularité de ces méthodes [189]. D'autre part, par le fait qu'elles fonctionnent au niveau client, ce qui facilite leur implémentation et favorise leur passage à l'échelle [181].

Méthode	Positionnement	Encodage	Complexité	Compatibilité avec le flux temps réel	Passage à l'échelle
Transcodage évolutif	Coté serveur	Petit	- -	Oui	-
Encodage évolutif	Coté serveur	Petit	-	Oui	-
Qualité variable	Coté client	Grand	+	Oui	+

TABLE 5.1 – Comparaison des stratégies d'adaptations de service vidéo

5.2.2 Diffusion vidéo en flux adaptatif dynamique sur HTTP

À partir de 2011, une nouvelle norme appelée "diffusion de flux vidéo adaptatif et dynamique sur HTTP" ("*DASH*" (*Dynamic Adaptive Streaming over HTTP*)), s'est imposée comme un standard dans la catégorie des stratégies basées sur le changement de qualité (Section (5.2.1)). Pour les grands acteurs du secteurs du web (*Microsoft, Adobe, Apple, Google, etc.*),

le but est de permettre d'assurer l'interopérabilité entre les différents produits ou systèmes existants sans restriction d'accès ou de mise en œuvre.

Dans le système Dash, le fichier vidéo est divisé en plusieurs segments (fragments) de même durée (habituellement de 2 à 10 secondes) et chaque segment vidéo est encodé au niveau du serveur avec plusieurs niveaux discrets de débit binaire [66]. Le standard Dash fonctionne sur le principe de changement dynamique de la représentation des segments vidéo. Pour éviter les problèmes d'arrêt dus à des raisons multiples (comme une faible bande passante ou un tampon vide, etc.), ce changement est réalisé sur la base des différents paramètres, qui sont mesurés passivement au niveau de l'application du côté client tels que la bande passante, le tampon vidéo, les propriétés de l'appareil, etc. La Figure (5.1) montre un exemple typique de déploiement d'un système Dash.

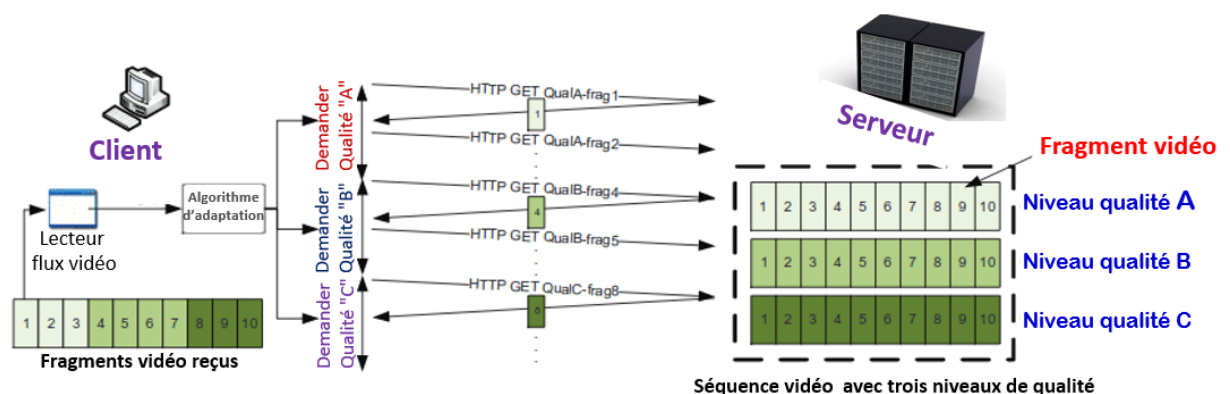


FIGURE 5.1 – Exemple d'un déploiement typique d'un système Dash

Le déroulé d'une session est comme suit : un client utilise un lecteur vidéo (Dash) compatible avec le navigateur Web. Il souhaite regarder une vidéo composée de 10 segments qui sont, chacun, encodés en 3 niveaux de qualité ("représentation"). Pendant que le client regarde un segment vidéo, l'algorithme d'adaptation de qualité (ABR) du lecteur vidéo met à jour le niveau de qualité recommandé en fonction des métriques mesurées par le client, ici, il s'agit du débit réseau. Le passage d'une qualité à une autre est décidé en fonction de la variation

de cette mesure de débit. Une séquence vidéo, peut donc être construite d'un assemblage de segments de différentes qualités. *Adobe Dynamic Streaming*⁴, *Microsoft Smooth Streaming* (MSS⁵) et *Apple HTTP Live Streaming* (HLS⁶) sont des implémentations populaires de Dash. Plusieurs études utilisant la norme Dash ont été proposés. Celles-ci peuvent être classées en trois catégories : les approches basées sur le débit (TBA) [108, 42, 75, 184], les approches basées la mémoire tampon (buffer) (BBA) [170, 85, 172, 44, 189] et les approches hybrides [82, 53, 117, 81, 179, 181, 167, 173, 109, 115, 156, 77]. Nous présentons ci-dessous certaines de ces études.

i) Approches basées sur le débit :

Dans [108], Azumi et al. ont introduit une nouvelle méthode ABR de type TBA pour la diffusion vidéo à débit variable. Cette étude a permis de confirmer que l'ajout d'un niveau de qualité intermédiaire est un moyen efficace d'améliorer la QoE usager, en particulier lorsque les conditions réseaux changent d'une manière fréquente (environnements mobiles).

Dans [42], Liu et al. ont utilisé l'analyse statistique pour proposer une méthode ABR. Une logique empirique est utilisée pour calculer la bande passante disponible à partir du temps de téléchargement du segment précédent et de la dernière représentation vidéo utilisée.

Dans [75], Jiang et al. ont analysé plusieurs acteurs du marché et ont proposé une solution nommée "FESTIVE" afin d'optimiser l'estimation de la bande passante réseau. Grâce à cette solution, les auteurs ont identifié les causes des problèmes de la diffusion vidéo qui surviennent à la suite de l'utilisation d'un grand nombre de qualités intermédiaires.

Dans [184], Sun et al. ont développé un outil nommé "CS2P". Cet outil se base sur un modèle de chaînes de markov cachées (*Hidden Markov Model (HMM)*) afin de modéliser la variation de la qualité en fonction de l'évolution du débit dans le temps.

4. <http://www.adobe.com/devnet/hds.html>

5. <https://www.iis.net/downloads/microsoft/smooth-streaming>

6. <https://developer.apple.com/streaming/>

ii) Approches basées sur le mémoire tampon :

Dans [85], Miller et al. ont présenté une méthode dont l'objectif est d'éviter l'épuisement de la mémoire tampon. Les résultats ont démontré que ce type d'approche est particulièrement adapté aux réseaux cellulaires.

Dans [170], Huang et al. ont introduit une approche qui se base sur l'évolution de la mémoire tampon. Les auteurs ont testé cette approche dans le cadre d'un déploiement dans une plateforme réelle en partenariat avec "Netflix". Les résultats montrent une nette amélioration des performances par rapport aux méthodes basées sur l'estimation du débit réseau. Cet écart est encore plus important lorsque les conditions réseaux varient fréquemment.

Dans [172], Krishnamoorthi et al. ont proposé une approche basée sur l'apprentissage automatique. Cette méthode, appelée *BUFFEST*, permet de prédire et de classer l'état de la mémoire tampon au niveau du client à partir du trafic HTTP et HTTPS.

Dans [44], Zhou et al. ont proposé une méthode qui utilise deux seuils (un seuil de débordement et un seuil de débit minimal) pour gérer la mémoire tampon. Ainsi, ils proposent d'estimer la variation de la bande passante en fonction du taux de remplissage de la mémoire tampon. L'objectif est d'adapter en continue la qualité de la vidéo en utilisant ces seuils dont le but d'éviter les interruptions de lecture.

Dans [189], Zahran et al. ont proposé une approche nommée "OSCAR" (Optimized Stall-Cautious Adaptive bitRate). À l'image de la proposition [44], cette approche utilise 3 niveaux de mémoire tampon qui sont calculés d'une manière expérimentale. Le but de cette approche est d'estimer les débit réseau et choisir la meilleure qualité vidéo selon le niveau de la mémoire tampon et le débit estimé.

iii) Méthodes hybrides :

Dans [77], Liu et al. ont proposé une approche centrée sur l'utilisateur (QoE) pour la livraison de vidéos streaming dans les réseaux de distribution de contenus (CDNs). L'objectif de ce travail est de maximiser la QoE en termes de score MOS des téléspectateurs utilisant l'infrastructure "CDN". L'intérêt de cette étude consiste dans l'intégration de la solution dans une infrastructure réelle.

Dans [82], Juluri et al. ont introduit une nouvelle méthode appelée *SARA* pour assurer la diffusion adaptative de la vidéo. Cette méthode calcul le temps requis pour télécharger le segment vidéo suivant. Elle prend en considération de nombreux facteurs, y compris : la variation de taille de segment, la bande passante estimée et l'occupation de la mémoire tampon.

Dans [117], Mushtaq et al. ont proposé une nouvelle approche qui permet de sélectionner dynamiquement la représentation vidéo appropriée en fonction des conditions du réseau (bande passante réseau) et les indicateurs du lecteur vidéo (temps mémoire tampon et taux de perte d'images par seconde). La performance de la méthode proposée a été évaluée et comparée selon la méthode standard d'Adobe (*Adobe Dynamic Streaming*), mais les auteurs n'ont pas considéré l'impact combiné de trois paramètres sur la qualité de la vidéo perçue par les utilisateurs finaux.

Dans [81], Joseph et al. ont proposé une optimisation de la norme Dash (Figure (5.1)) pour le service vidéo à la demande. L'objectif de l'étude est de maximiser la QoE et de satisfaire le maximum nombre d'utilisateurs instantanés. En effet, l'idée consiste à trouver un compromis entre la qualité vidéo moyenne affichée, la variabilité temporelle de cette qualité et l'équité entre les utilisateurs, en intégrant leurs préférences sur le coût du service.

Dans [179], Xiang et al. ont présenté le problème de diffusion adaptative de la vidéo comme un problème de chaînes de markov. Ils ont utilisé la programmation dynamique pour obtenir la politique de diffusion optimale. Pour cela, ils ont utilisé plusieurs facteurs, tels que le temps d'interruption de la vidéo, la qualité moyenne affichée et la qualité instantanée de la vidéo. Pour réduire la complexité de la politique proposée, les auteurs ont introduit un algorithme en ligne qui se base sur les statistiques de la bande passante réseau et qui choisi les profils des prochains segments.

Dans [181], Chen et al. ont formulé mathématiquement une approche sous forme d'un processus de décision markov (MDP) pour étudier les causes qui poussent les télé-spectateurs à arrêter la vidéo. Le but est d'optimiser le temps de vision des spectateurs en assurant une bonne qualité de la vidéo affichée. Les résultats de ce travail ont démontré que l'utilisation d'une méthode hybride (BBA et TBA) incluant plusieurs mesures (telles que : la bande passante réseau, l'état de la mémoire tampon vidéo, le temps de la mise en tampon) pourrait améliorer considérablement la QoE des utilisateurs finaux de la vidéo.

Dans [167], Zhang et al. ont proposé une nouvelle méthode de diffusion vidéo adaptative de type hybride. Cette méthode se base sur l'analyse stochastique. Elle analyse, d'une part, l'influence de la variation de la taille de la mémoire tampon sur la probabilité d'avoir un tampon vide, et d'autre d'autre part, elle étudie l'impact du niveau de débit moyen sur les algorithmes de diffusion vidéo. Cette étude fournit aussi plusieurs recommandations pour la conception d'algorithme et l'encodage des segments vidéo.

Dans [173], Menkovski et al. ont introduit l'utilisation de l'apprentissage par renforcement pour la diffusion adaptative de la vidéo dans le but d'optimiser la satisfaction usager. En effet, des facteurs subjective de perception humaine ont été pris en considération dans le choix de la représentation des segments vidéo.

Dans [109], Claeys et al. ont proposé aussi une solution de diffusion de vidéo adaptative en utilisant l'apprentissage par renforcement. Cette solution est basée sur un algorithme de type "Q-Learning". Le but est de permettre au lecteur vidéo de découvrir dynamiquement le comportement optimal correspondant à chaque contexte afin de maximiser la QoE de l'utilisateur.

Dans [115], Mc Lean et al. ont développé une nouvelle méthode de diffusion vidéo à débit variable appelée : *Apple HTTP Live Streaming (HLS*⁷). Cette méthode est mise en œuvre par Apple Inc. et elle est implémentée dans ses logiciels *QuickTime*, *Safari*, et *iOS*. Elle est également intégrée dans les navigateurs web les plus connus comme *Firefox* et *Google Chrome*.

Dans [156], Spiteri et al. ont présenté une nouvelle méthode de diffusion vidéo adaptative nommée : BOLA. Cette méthode se base sur l'utilisation du problème de maximisation de l'utilité et utilise les techniques d'optimisation de Lyapunov [114] pour minimiser l'épuisement de la mémoire tampon et maximiser la qualité vidéo. Cette solution est implémentée dans le lecteur "Dash.js". Celui-ci est développé à l'initiative du "*Dash Industry Forum*" et consiste en un lecteur vidéo qui lit le contenu "MPEG-DASH" à l'aide de bibliothèques JavaScript.

Le Tableau (5.2) résume quelques études qui ont été réalisées dans la littérature pour proposer des méthodes ABRs en utilisant le standard Dash.

7. <https://developer.apple.com/streaming/>

Solution	Catégorie ABR	Modèle mathématique	Sensible à la QoE	Nom
[42]	Basée sur le débit	Analyse statistique	Non	/
[108]	Basée sur le débit	Analyse statistique	Oui	/
[75]	Basée sur le débit	Analyse statistique	Non	FESTIVE
[184]	Basée sur le débit	Chaîne de Markov	Oui	CS2P
[170]	Basée sur le tampon	Analyse statistique	Non	/
[85]	Basée sur le tampon	Analyse statistique	Non	/
[172]	Basée sur le tampon	Apprentissage automatique	Non	BUFFEST
[44]	Basée sur le tampon	Analyse statistique	Non	/
[189]	Basée sur le tampon	Analyse statistique	Non	OSCAR
[82]	Hybride	Analyse statistique	Non	SARA
[117]	Hybride	Analyse statistique	Oui	BBF
[81]	Hybride	Analyse statistique	Oui	NOVA
[177]	Hybride	Analyse statistique	Oui	/
[179]	Hybride	Chaîne de Markov	Non	/
[181]	Hybride	Chaîne de Markov	Oui	/
[167]	Hybride	Analyse stochastique	Non	/
[115]	Hybride	Analyse statistique	Non	HLS
[156]	Hybride	Analyse statistique	Non	BOLA
[145]	Hybride	Programmation Dynamique	Oui	/
[163]	Hybride	Programmation Dynamique	Oui	/
[173]	Hybride	Apprentissage par renforcement	Oui	/
[109]	Hybride	Apprentissage par renforcement	Oui	/

TABLE 5.2 – Méthodes diffusion vidéo à débit variable en utilisant le standard Dash

5.3 Campagne de test subjective

Pour mettre en œuvre le modèle CBF de calcul de la QoE présenté dans le Chapitre (4), nous avons besoin d'une base de données subjective pour le service vidéo qui utilise la norme Dash. La construction de telle base de données est une tâche difficile, vu la nécessité de mobiliser des ressources humaines et matérielles, en plus du temps et du coût financier. Notre objectif est de construire une base de données qui servira à nos différentes expérimentations et qui sera mise à disposition du public afin d'être utilisée par des chercheurs d'autres institutions, pour les comparaisons et les améliorations des modèles de calcul de la QoE usager et leurs usages dans des services qui implémentent la norme Dash.

5.3.1 Construction de la base de données

Dans notre campagne de test, une base de données subjective a été construite avec la méthode *Note Absolue par Catégorie (ACR)*[88, 2, 7]. Le score d'évaluation MOS (Mean Opinion Score) a servi à quantifier la QoE perçue chez les usagers.

Les tests ont été effectués dans des environnements filaires (câblés) et sans fil. Tous les dispositifs d'affichage sont compatibles avec la norme Dash et sont connectés soit au réseau filaire (ordinateurs de bureau/ ordinateurs personnels) ou soit à un point d'accès Wifi (téléphones mobiles : Samsung S5, HTC X One, Samsung 4 Mini et Samsung S3). Ces dispositifs ont été sélectionnés car ils ont été les dispositifs Android les plus vendus en 2014 en France⁸. En plus, ils sont représentatifs des dispositifs utilisés par le grand public à cause de la diversité de leurs caractéristiques techniques telles que la mémoire, la résolution d'écran, la taille de la fenêtre d'affichage, etc. La Figure (5.2) présente la description générale de la campagne réalisée.

En résumé, cette campagne de test prend en compte plusieurs aspects suggérés par les recommandations de l'IUT [2] et du groupe QUALINET [160]. Nous citons ci-dessous quelques-uns.

- 1) **Affichage** : Une application Javascript dédiée basée sur le lecteur vidéo "Dash.js"⁹ a été développée pour l'expérimentation. L'ordre d'apparition des séquences vidéo

8. <http://www.leparisien.fr/high-tech/apple-et-samsung-en-tete-du-top-10-des-meilleures-ventes-mondiales-de-smartphones-17-07-2014-4010539.php>

9. <http://dashif.org/>

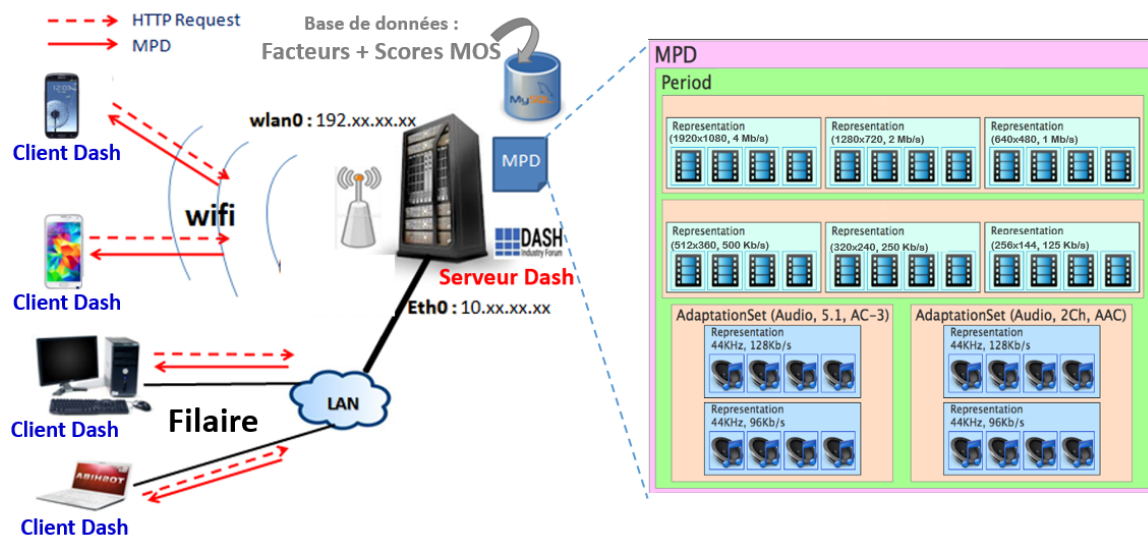


FIGURE 5.2 – Campagne de test réalisée pour la mise en application du modèle CBF

est aléatoire. Le lecteur vidéo développé propose des vidéos avec différentes représentations vidéo, comme présenté dans le fichier "MPD" (Figure 5.2). À noter que lors des expériences, la règle d'adaptabilité par défaut du lecteur vidéo "Dash.js" a été désactivée, dans le sens où une seule représentation vidéo est sélectionnée pour l'ensemble des segments d'un test. En pratique, la page principale de notre application de test contient le lecteur vidéo et deux boutons : un pour démarrer la séquence vidéo et le deuxième pour fournir les retours des utilisateurs en termes de valeurs MOS. Le score MOS et les paramètres mesurés (les conditions de réseau, les paramètres vidéo et les caractéristiques du dispositif d'affichage) sont stockés dans la base de données MySQL à l'aide d'un script PHP. Les conditions techniques du réseau sont émulées pendant les tests en utilisant l'émulateur de réseau NetEm [147].

2) Participants : 42 personnes ont participé aux tests. Ces personnes sont des étudiants et des chercheurs du laboratoire *LiSSi*. Leur âge est compris entre 21 et 41 ans avec peu ou pas d'expérience dans l'évaluation de la vidéo. 20% des participants ont vu au moins 7 vidéos pendant une session de test et 23% des participants sont des femmes.

3) Questions : Les questions sont simples comme suggéré par le rapport du réseau QUALINET pour faciliter la procédure d'expérimentation. Exemple : *Veillez noter la qualité globale de la vidéo.*

4) Réplication des séquences vidéo : Les mêmes séquences vidéo et les mêmes paramètres sont présentés à plusieurs testeurs. Le nombre de séquences utilisées est de 2 et chacune a une durée de 60 secondes. Elles ont été extraites de la vidéo originale "Big Buck Bunny"¹⁰ couramment utilisée dans ce genre de test. Les deux séquences vidéo sont encodées dans les mêmes débits binaire vidéo à l'aide du codec H.264. Le Tableau (5.3) présente les différentes représentations utilisées.

Représentation	Débit binaire vidéo	Résolution	Images/sec	Codec
6(1080p30)	4 Mb/s	1920 x 1080	15/25/30	H264
5(720p30)	2 Mb/s	1280 x 720	15/25/30	H264
4(480p30)	1 Mb/s	640 x 480	15/25/30	H264
3(360p30)	500 Kb/s	512 x 360	15/25/30	H264
2(240p30)	250 Kb/s	320 x 240	15/30	H264
1(144p10)	125 Kb/s	256 x 144	10	H264

TABLE 5.3 – Représentations utilisées dans le fichier *MPD*

Durant la campagne de test, nous avons construit la base de données [96] qui contient 300 exemples. Chaque exemple contient 7 facteurs pouvant impacter la QoE dont 3 issus du réseau (QoS), 3 de la vidéo (QoA) et un paramètre relatif au périphérique utilisé. La description des facteurs utilisés est présenté dans le Tableau (5.4) ci-dessous. La colonne "Catégorie" représente la classe d'appartenance des facteurs utilisés (Section 1.4)

10. <https://www.youtube.com/watch?v=aqz-KE-bpKQ>

Facteur	Catégorie	Description
<i>BWD</i>	<i>QoS</i>	La capacité moyenne du réseau en termes de "bits/s".
<i>delai</i>	<i>QoS</i>	La latence du réseau lors du test en termes "ms".
<i>gigue</i>	<i>QoS</i>	La variation de la latence au fil du temps en termes "ms".
<i>qualite</i>	<i>QoA</i>	La représentation affichée (144p10, , 1080p30). Voir Tableau (5.3)
<i>FR</i>	<i>QoA</i>	Le nombre d'images projetées ou affichées par seconde.
<i>Floss</i>	<i>QoA</i>	Le nombre d'images perdues par seconde.
<i>CPUmoy</i>	<i>QoD</i>	Le taux d'utilisation du CPU (%).
<i>MOS</i>	<i>mos</i>	Valeur du MOS (Mean Opinion Score).

TABLE 5.4 – Facteurs collectés pour mettre en application de notre proposition CBF

5.3.2 Sélection des facteurs

Nous avons utilisé notre approche, présentée dans la Section (3.2), pour sélectionner les facteurs les plus pertinents lors du calcul de la QoE perçue par l'utilisateur. Cette approche se base sur l'étude statistique de la corrélation de Pearson " r ". Au début, nous calculons la matrice de corrélation de Pearson " r " pour les 7 facteurs présentés dans le Tableau (5.4). Cette matrice est présentée dans la Figure (5.3).

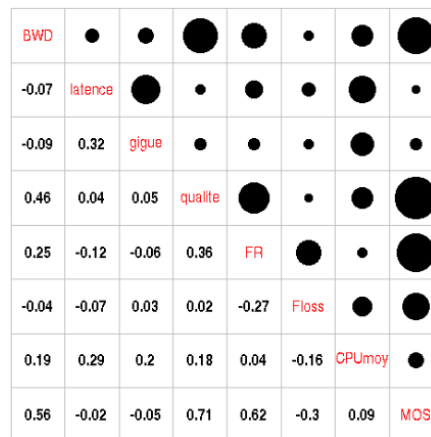


FIGURE 5.3 – Matrice de corrélation utilisée pour l'application du modèle CBF

Le premier facteur sélectionné est la représentation vidéo (*qualite*), car c'est le facteur ayant le plus grand taux de corrélation ($r = 0.71$) par rapport au MOS. Les autres étapes d'exécution de notre approche sont présentées dans le Tableau (5.5). La deuxième colonne de ce tableau concerne les facteurs sélectionnés. La troisième colonne quant à elle présente les facteurs candidats avec leur valeur ajoutée calculée (équation 3.6). Les autres colonnes présentent les performances de prédiction en utilisant la meilleure méthode d'apprentissage automatique obtenue (boosting avec le gradient (*BoostGrad*) présenté dans la Section 3.3.2).

Étape	facteurs sélectionnés	facteurs candidats	r	MAE	RMSE
02	<i>qualite</i>	<i>Floss</i> ($V = 1.28$)	0.901	0.402	0.561
03	<i>qualite</i> et <i>Floss</i>	<i>BWD</i> ($V = 1.52$)	0.955	0.367	0.489
04	<i>qualite</i> et <i>Floss</i> et <i>BWD</i>	<i>FR</i> ($V = 1.37$)	0.9528	0.273	0.401
04	<i>qualite</i> et <i>Floss</i> et <i>BWD</i>	<i>Gigue</i> ($V = 0.96$)	0.943	0.307	0.421
04	<i>qualite</i> et <i>Floss</i> et <i>BWD</i>	<i>Delais</i> ($V = 0.95$)	0.942	0.312	0.429
04	<i>qualite</i> et <i>Floss</i> et <i>BWD</i>	<i>CPUmoy</i> ($V = 0.90$)	0.936	0.316	0.432

TABLE 5.5 – Performances de l'approche de sélection des facteurs

Dans le Tableau (5.5), nous observons clairement que dans les étapes 2 et 3, le premier facteur candidat (ayant la plus grande valeur ajoutée calculée par l'équation (3.6)) donne un bon résultat. Par conséquent, il est sélectionné directement à chaque fois. À partir de l'étape 4, aucun facteur issu de l'ensemble des facteurs candidats n'améliore les performances de calcul (taux de corrélation, erreur absolue moyenne et erreur quadratique moyenne). De ce fait, la liste finale des facteurs sélectionnés sera limitée aux trois facteurs déjà sélectionnés : La représentation vidéo (*qualite*), le taux de perte d'images par seconde (*Floss*) et la bande passante réseau (*BWD*). Le taux de corrélation à l'issue du déroulement de la méthode de sélection des facteurs pertinents dépasse les 0.955. Le taux moyen d'erreur résultant en termes d'erreur absolue moyenne est de 0.37 et de 0.49 en termes d'erreur quadratique moyenne.

5.3.3 Construction du modèle

Pour commencer la construction de notre modèle de calcul CBF, nous avons choisi dans la première étape de ce modèle les trois facteurs sélectionnés par notre approche dans la section précédente.

Dans la deuxième étape du modèle CBF, nous avons utilisé comme oracles (sources d'information) les méthodes d'apprentissage automatique suivantes : les réseaux de neurones multicouches (*RNN*), les machines à vecteurs de support (*SVM*), les forêt aléatoire (random forest (*RF*)), le bagging avec arbres de décision (*BaggingDT*), le boosting avec le gradient (*BoostGrad*) et le boosting avec arbre de décision (*BoostDT*). Ces différentes méthodes ont été explicitées dans la Section (3.3) du Chapitre 3. Une validation croisée avec dix échantillons a été utilisée pour choisir la meilleure configuration de chaque méthode d'apprentissage automatique comme réalisée dans la Section (3.4).

Dans la troisième étape du modèle CBF, les valeurs calculées par les différentes méthodes d'apprentissage automatique (étape 2) sont utilisées par le modèle CBF. La meilleure règle sélectionnée dans le chapitre précédent *Murphy pondérée* (notée : *CBFmurphy*), a été utilisée pour calculer la QoE usager.

Le Tableau (5.6) résume les résultats de calcul obtenus en utilisant les méthodes d'apprentissage automatique et le modèle CBF (*CBFmurphy*).

Nom	Acronyme	Performances de prédiction			Temps de calcul
		MAE	RMSE	r	
Régression linéaire	<i>LM</i>	0.736	0.850	0.79	0.001 (ms)
machines à vecteurs de support	<i>SVM</i>	0.316	0.450	0.93	0.003 (ms)
Forêts randomisées	<i>RF</i>	0.305	0.411	0.94	0.002 (ms)
Boosting avec le gradient	<i>BoostGrad</i>	0.267	0.389	0.95	0.010 (ms)
CBF avec règle Murphy pondérée	<i>CBFmurphy</i>	0.245	0.352	0.96	0.025 (ms)

TABLE 5.6 – Tableau comparatif des méthodes d'estimation de la QoE

La troisième colonne de ce tableau concerne les performances de prédiction : l'erreur quadratique moyenne (RMSE), l'erreur absolue moyenne (MAE) et le coefficient de corrélation (r). La dernière colonne quant à elle concerne le temps de calcul de chaque méthode en millisecondes (ms).

Nous constatons ici qu'à l'exception de la régression linéaire (modèle simple), l'utilisation des méthodes d'apprentissage automatique produit de bons résultats pour calculer la QoE perçue par l'utilisateur sur la base de données issue de notre étude. Comme nous pouvons le constater dans le Tableau (5.6), notre proposition construite grâce à la méthode CBF avec la règle de fusion "*Murphy pondérée*" obtient les meilleures performances. L'erreur absolue moyenne est de 0.24 et le taux de corrélation entre les valeurs de la QoE prédites et les valeurs réelles de la QoE est de 0.96. Ces résultats nous ont donc amené à sélectionner cette méthode pour la construction de notre solution de diffusion vidéo à débit variable qui sera présentée dans la partie suivante.

5.4 Solution proposée de diffusion vidéo à débit variable

Pour mettre en application notre modèle de calcul de la QoE (CBF), nous proposons dans cette section une nouvelle solution de diffusion à débit variable (ABR). Cette solution est notée : "Q2ABR" (*Quality of Experience aware Adaptive Bit Rate*). Elle utilise l'apprentissage par renforcement pour la sélection de la représentation vidéo des segments en fonction de la variabilité de la QoE calculée. L'objectif ici est de proposer une nouvelle méthode de diffusion de séquences vidéos à débit variable sensible à la QoE et permettant d'optimiser l'utilisation de la bande passante. Cette solution est décomposée en deux sous-modules : i) un module de calcul de la QoE et ii) un modèle paramétrique qui optimise en temps réel la sélection des représentations vidéo en fonction de la mesure instantanée de la QoE calculée. La Figure (5.4) présente la description générale de l'architecture de notre solution "Q2ABR". Dans cette architecture, le module "*calcul QoE*" concerne le premier module qui se charge de calculer les valeurs QoE perçues par l'utilisateur. Les deux sous-modules "*Détection points de changement QoE (BP-QoE)*" et "*Sélection de qualité (AgentABR)*" composent le modèle paramétrique qui optimise en temps

réel la qualité de la vidéo en fonction de la mesure instantanée de la QoE calculée. Ces deux modules sont présentés ci-après.

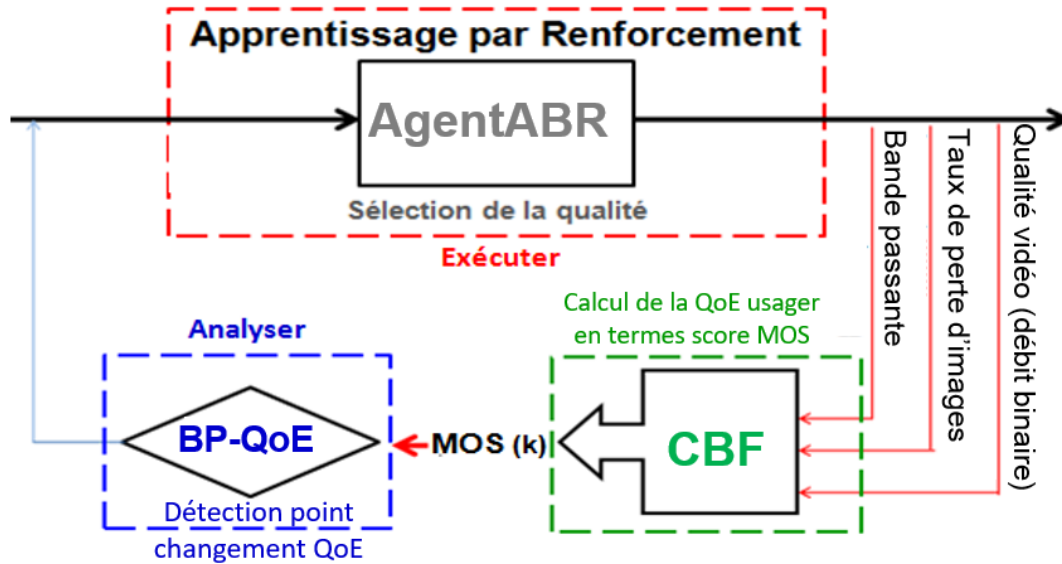


FIGURE 5.4 – Architecture générale

5.4.1 Surveillance de changement de la QoE

Le système de diffusion vidéo est impacté par plusieurs facteurs qui varient dans le temps et qui ont une influence différente sur la qualité perçue par l'utilisateur (QoE). Par exemple, pour certains facteurs comme le débit binaire vidéo ou la bande passante réseau, l'augmentation impacte positivement la QoE perçue par l'utilisateur. D'autres facteurs comme le délai de bout en bout ou le nombre d'images perdues, l'impactent négativement.

Pour quantifier les variations des valeurs de la QoE dans le temps, les changements de valeurs instantanées de ces facteurs sont mesurés. La surveillance de ces variations est un point très important dans le système de diffusion vidéo [108, 145]. Elle permet de savoir quand déclencher des actions spécifiques correctrices suite à l'arrivée d'événements imprévisibles ou non souhaités comme une QoS insatisfaisante, un retour d'expérience négatif ou encore des dysfonctionnements des éléments du réseau.

Pour cela, l'idée maîtresse dans le cadre des travaux de cette thèse est de surveiller la variation de la QoE usager sur la base des scores MOS calculés à partir des facteurs instantanés mesurés. Après avoir évalué un ensemble de solutions, nous avons finalement opté pour l'utilisation du mécanisme de "détection de point de changement" (*Break point detection*) [91]. Ce mécanisme permet de localiser les grandes variations dans le temps au sein d'un jeu de données. Il concerne tous les systèmes dépendants du temps et qui utilisent une séquence de variables aléatoires décrite par un nombre fini de paramètres structurels [135]. Comme exemple, nous pouvons citer l'identification des signaux électriques transitoires précédant les tremblements de terre (caractérisée par une dynamique critique). Ces signaux peuvent être distingués des signaux semblables émis par les sources de bruit proches en utilisant le mécanisme de détection de point de changement.

Dans la théorie de l'information, il existe plusieurs types de méthodes pour détecter les changements. On peut classer celles-ci en deux catégories [46]. La première consiste dans les tests pour les lois unidimensionnelles où les ruptures potentielles peuvent survenir uniquement via un changement d'un seul paramètre (θ), tandis que la deuxième représente les tests pour les lois multidimensionnelles qui consistent à élaborer des points de changement dans une série d'observations multivariées (plusieurs variables). Dans cette thèse, nous nous focalisons sur la première catégorie. Des tests de suivi de changement de la QoE en termes de scores MOS sont réalisés pour définir les points de changements de la QoE (notés "BP-QoE"). Pour modéliser ces derniers, considérons une suite d'observations, dans notre cas les scores MOS instantanés $MOS = (MOS_1, MOS_2, \dots, MOS_L)$, de longueur L que nous calculons périodiquement avec une méthode d'estimation *CBF* (Chapitre (4)). Le MOS est réparti indépendamment dans le temps (système dynamique i.e. facteurs variables dans le temps). La Figure (5.5) modélise cet exemple.

Le mécanisme de détection de point de changement consiste alors à faire une segmentation de la séquence des scores MOS , en détectant (i) les positions des points de changement $C = (c_1, c_2, \dots, c_N)$, où : $1 = c_0 < c_1 < \dots, c_{N+1} < L + 1$ (L nombre d'observation) et (ii) le nombre de points de changement N qui représentent les grandes variations des valeurs calculées. Ainsi, un point de changement (c_i) est la limite entre deux segments adjacents ayant

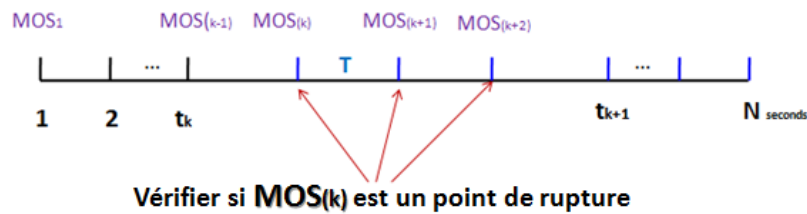


FIGURE 5.5 – Modélisation des points de changement

chacun sa propre distribution des valeurs de MOS calculées.

Afin de détecter les points de changement ("BP-QoE") en utilisant les tests pour les lois unidimensionnelles, plusieurs techniques existent dans la littérature. Toutefois et d'après [67], déterminer l'existence d'un changement est d'autant plus difficile que ce dernier n'est pas forcément caractérisé par un décalage de grande amplitude entre deux valeurs successives (MOS_k et MOS_{k+1}) par rapport à la dispersion des observations. Un enjeu de la détection est donc d'être sensible aux faibles variations tout en garantissant une certaine robustesse au bruit. Par conséquent, nous avons utilisé la technique de minimisation de l'entropie croisée (CE)[139, 134], qui est une technique d'optimisation stochastique basée sur une méthode de recherche exacte, pour estimer à la fois le nombre et l'emplacement des changements dans les données d'*hybridation génomique comparative*[39]. Cette méthode se base sur une procédure d'optimisation stochastique basée sur les distributions de probabilités et elle est disponible sous forme de paquet [13] dans la boîte à outils *R* software[1].

Dans notre cas, la méthode de l'entropie croisée (CE)[139, 134] est implémentée dans le premier sous module ("BP-QoE" dans la Figure (5.4)) de notre solution ("Q2ABR") de diffusion vidéo à débit variable. Son but est de surveiller les variations de la QoE en termes des valeurs du MOS calculés par le modèle *CBF* (Chapitre 4) et de signaler les points de changements détectés. Le résultat de cette méthode est utilisée ensuite par le deuxième sous module ("sélection de qualité vidéo "AgentABR")" dans la Figure (5.4)) de notre modèle paramétrique. La partie suivante présente le modèle formel du fonctionnement de ce dernier.

5.4.2 Modèle formel

La représentation vidéo affichée aux utilisateurs de la vidéo doit être adaptée selon le contexte d'utilisation (facteurs impactant la QoE usager). Ce mécanisme d'adaptabilité pour sélectionner un segment vidéo optimal peut être modélisé comme un problème d'apprentissage par renforcement (AR)[109, 173], qui consiste en un agent (noté "AgentABR") travaillant dans un environnement (Figure (5.6)) où il apprend les actions à prendre, à partir d'observations, de façon à optimiser une récompense quantitative au cours du temps.

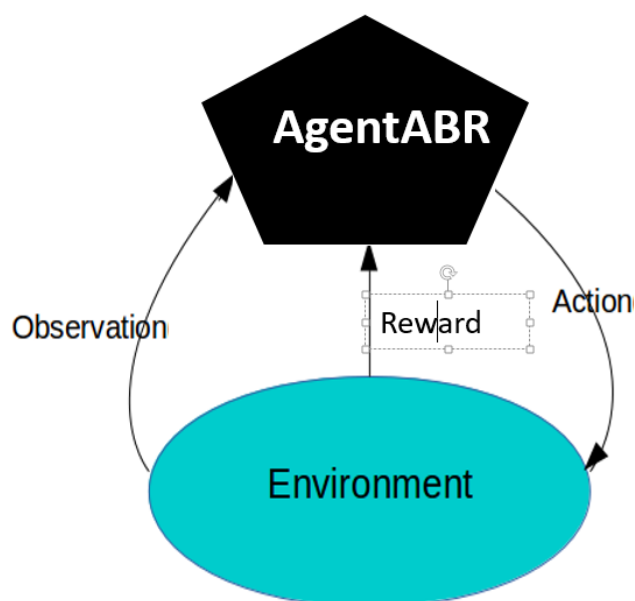


FIGURE 5.6 – Système apprentissage par renforcement

En effet, l'agent exécute des actions sur l'environnement et reçoit ensuite les changements d'état sous forme d'observation de cet environnement et les réponses de ce dernier[173]. En termes mathématiques, le tuple $\{S, A, T, R, \gamma\}$ définit les composantes du système d'apprentissage par renforcement (AR), où **S** définit les états, **A** présente les actions, **r** indique les récompenses du système et γ représente un paramètre de mise à jour qui sera utilisé par la politique de contrôle (noté π). Celle-ci est l'association d'une action a_i à chaque état s_j , ce qui maximise la récompense moyenne du gain (G) à long terme, défini par :

$$G = E^\pi \left[\sum_{i=1}^{\infty} (\gamma_i r_i) \right] \quad (5.1)$$

Dans notre approche, nous représentons un état ($\mathbf{s}_j \in \mathbf{S}$) comme une représentation vidéo et nous considérons trois types d'actions :

- (1) incrémente la représentation demandée ($\mathbf{a}_1 = \text{quality} ++$),
- (2) décrémente la représentation demandée ($\mathbf{a}_2 = \text{quality} --$),
- (3) invariante (utiliser la même représentation vidéo $\mathbf{a}_3 = \text{quality}$).

La fonction d'apprentissage du processus de sélection de la qualité vidéo consiste à sélectionner une action a_i pour passer d'un état à un autre (si nécessaire) pour le segment vidéo suivant à demander. En conséquence, pour sélectionner a_i parmi toutes les actions possibles $A = \{a_1, a_2, a_3\}$, nous définissons :

- La fonction de récompense r comme l'estimation actuelle de la QoE (score MOS calculé avec la méthode CBF),
- La pénalité γ dépend des variations des MOS prédits. Cette valeur est calculée en utilisant la moyenne des scores calculés depuis le dernier point de changement et la variabilité de ces scores (l'écart type des valeurs MOS calculées). L'équation (5.2) présente ce calcul.

$$\begin{aligned} \gamma_{a_i} = & \text{abs}(\text{mean}(MOS_{\text{previous}}) - \text{mean}(MOS_{\text{present}})) \\ & + \text{abs}(\text{mean}(MOS_{\text{previous}}) - MOS_{a_i}) \end{aligned} \quad (5.2)$$

Avec :

- MOS_{a_i} : Valeur du MOS instantané estimée avec CBF pour l'action a_i .
- MOS_{previous} : Ensemble des valeurs des MOS prédites depuis le dernier point de changement des scores MOS (BP-QoE).
- MOS_{present} : Ensemble des valeurs des MOS prédits jusqu'à maintenant. Il est construit avec MOS_{previous} en ajoutant MOS_{a_i} valeur.
- $\text{abs}()$: Fonction de valeur absolue.

L'action sélectionnée (a_i) à chaque prise de décision est celle ayant le minimum de pénalité (γ_{a_i}). Elle permet de prendre une décision sur la représentation vidéo à demander pour le prochain segment vidéo selon le fonctionnement de la norme Dash, comme illustré dans la Figure (5.1).

5.4.3 Fonctionnement

La solution proposée dans le cadre de cette thèse est basée sur le standard Dash (Figure (5.1)). Son objectif est d'optimiser en temps réel la représentation vidéo des segments à afficher en fonction de la mesure instantanée de la QoE calculée en termes de score MOS. La Figure (5.7) détaille son fonctionnement.

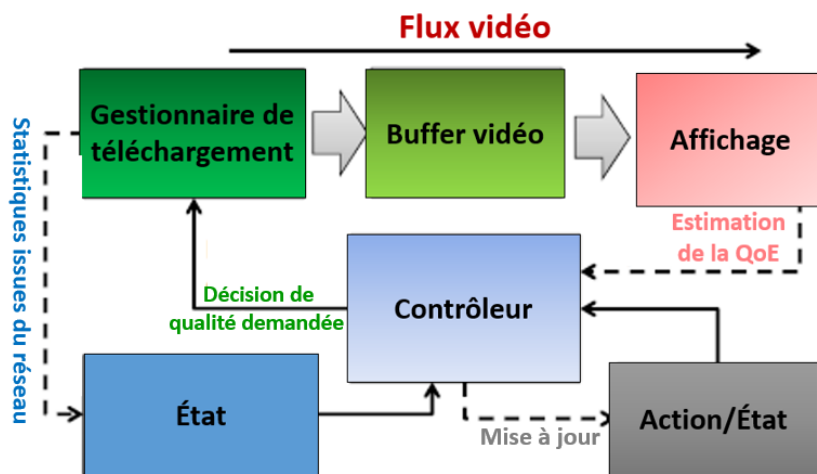


FIGURE 5.7 – Composantes de la solution

La solution se base sur l'utilisation de la méthode "CBF" pour le calcul de la QoE, couplée à un sous module de surveillance des changements de la QoE ("BP-QoE") et d'un agent intelligent (*AgentABR*) utilisé pour sélectionner la représentation vidéo des segments. À l'instar de la méthode "OSCAR"[189], la solution "Q2ABR" considère 3 niveaux de la mémoire tampon. Ils sont représentés comme suit :

$$0 < faible < \alpha_1 < transitoire < \alpha_2 < stable. \quad (5.3)$$

- α_1 et α_2 représentent deux bornes qui délimitent les trois zones et sont calculés d'une manière expérimentale.

Pour commencer, la représentation vidéo du premier segment est sélectionnée selon le débit réseau mesuré (bande passante) : la représentation ayant le débit binaire vidéo le plus proche de la moitié du débit mesuré est choisie. Exemple : Si le débit mesuré est égale à 1600 *Kb/s*, la représentation choisie sera la "4" (640x480, 1 Mb/s) avec le débit binaire vidéo de 1Mb/s (c.f. Tableau 5.3).

Le module CBF calcule périodiquement la QoE en termes de score MOS. À travers nos expérimentations, nous avons opté pour une période égale à la durée d'un segment vidéo, que nous avons fixé à 2 secondes comme le préconise la norme Dash [66]. Lorsque le score MOS est calculé, le sous-module de surveillance de la QoE ("BP-QoE") évalue l'existence d'un point de changement dans la qualité perçue par l'utilisateur. Si aucun point de changement n'est détecté, la même représentation vidéo est demandée pour le prochain segment, à condition que la mémoire tampon n'est pas remplie. Quand un point de changement ("BP-QoE") est détecté et que la mémoire tampon n'est pas remplie, l'*AgentABR* enclenche les actions nécessaires selon la formule (5.2) et l'état de la mémoire tampon.

- Si l'état de remplissage de la mémoire tampon est faible, la représentation vidéo inférieure est demandée, afin d'accélérer le remplissage du tampon. Autrement dit, les deux actions a_2 et a_3 sont privilégiées.
- Si l'état de remplissage de la mémoire tampon est élevée, le client privilégie une représentation vidéo supérieure tout en évitant l'épuisement de la mémoire tampon. Autrement dit, les deux actions a_1 et a_2 sont privilégiées.
- Au niveau transitoire, l'*AgentABR* teste les 3 actions définies auparavant et sélectionne la meilleure action (a_i) à partir du calcul de pénalité (γ_{a_i}) en utilisant l'équation (5.2).

Le diagramme présenté dans la Figure (5.8) résume le fonctionnement global de notre solution.

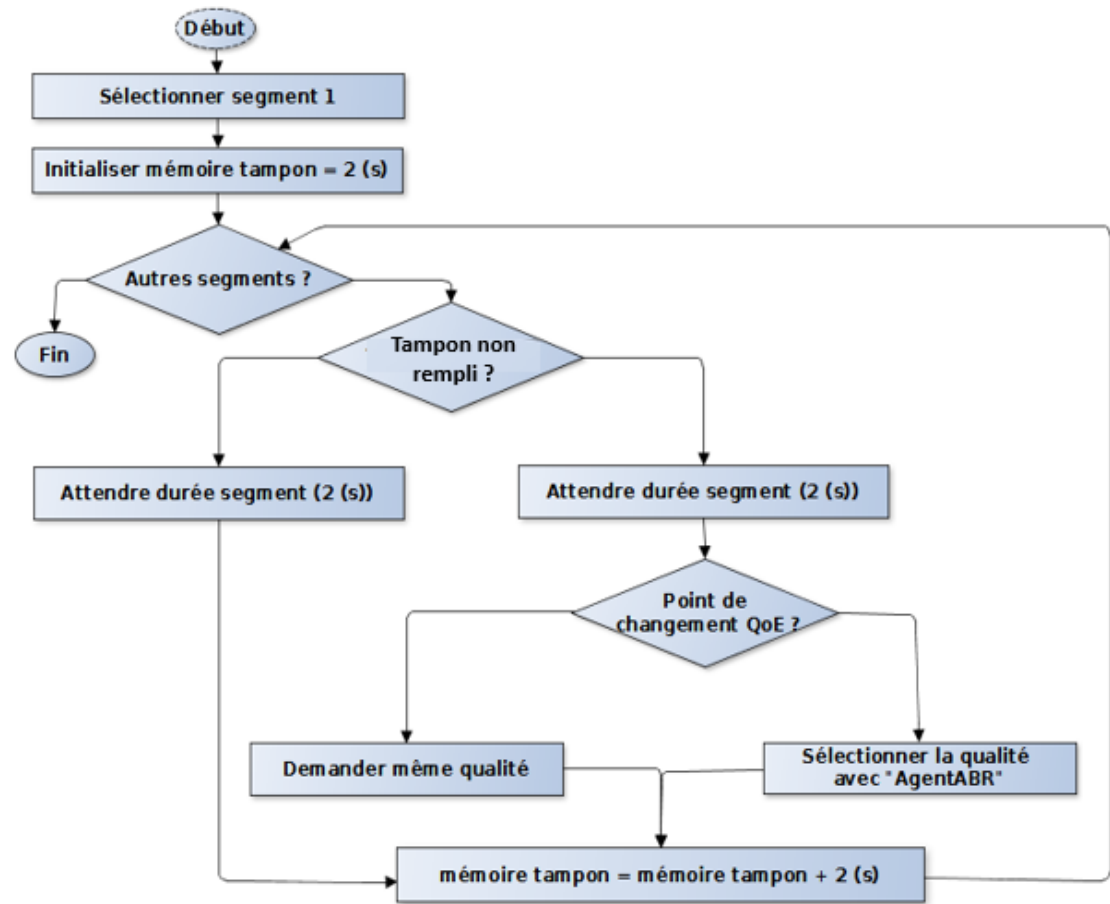


FIGURE 5.8 – Diagramme de fonctionnement de la solution

5.5 Évaluation

5.5.1 Solutions utilisées

Étant donné le grand nombre de facteurs qui impactent le système de diffusion vidéo en plus de sa contrainte de transfert de flux en temps réel, la mise en œuvre de solution de diffusion vidéo à débit variable est une tâche extrêmement difficile [145, 163]. Un certain nombre d'études ont été analysées dans la Section (5.2.2) afin d'obtenir une vue d'ensemble

de l'état de l'art. Parmi celles-la, nous avons sélectionné 3 solutions pour l'évaluation de notre méthode "Q2ABR". La première est la solution HTTP Live Streaming (HLS)[115]. Celle-ci a été sélectionnée car elle est implémentée par *Apple Inc.* et qu'elle est souvent utilisée comme un modèle de référence. La deuxième solution est l'algorithme *BOLA*[156], que nous avons sélectionné car il est intégré dans le lecteur vidéo "Dash.js" comme implémentation de référence. La dernière solution est "OSCAR"[189]. Celle-ci a été considérée à cause sa stratégie de gestion de la mémoire tampon dont on s'inspire. Le détail de ces méthodes est présenté ci-après :

— **Apple HTTP Live Streaming (HLS¹¹) :**

C'est une méthode de diffusion vidéo à débit variable mis en œuvre par *Apple Inc.* dans le cadre de ses logiciels *QuickTime*, *Safari*, et *iOS*. Elle est basée sur la norme Dash et elle est intégrée dans les navigateurs web les plus connus comme *Microsoft Edge*, *Firefox* et *Google Chrome*. Son principe de fonctionnement est résumé comme suit :

- Pour commencer, la représentation vidéo du premier segment est ajustée par rapport à la bande passante. Le débit binaire d'encodage utilisé est fixé à 70% de la valeur de la bande passante mesurée.
- Lorsque la mémoire tampon est vide, la représentation la plus basse est demandée.
- À chaque fin de téléchargement d'un segment, la valeur du taux de perte d'images par seconde est calculée. Cette valeur est surveillée pour diminuer la représentation vidéo utilisée lorsqu'elle dépasse les 20%.

— **Implémentation BOLA :**

La première étude introduisant la méthode BOLA est présenté par la méthode Spiteri et al. dans l'étude[156]. Cette méthode se base sur l'utilisation des techniques d'optimisation de Lyapunov [114] pour minimiser l'épuisement de la mémoire tampon et maximiser la qualité vidéo selon la norme Dash. L'initiative "*Dash Industry Forum*" a ensuite implémenté cette solution dans le lecteur "Dash.js". Pour sélectionner la meilleure re-

11. <https://developer.apple.com/streaming/>

présentation vidéo à demander, l'algorithme BOLA surveille le niveau de la mémoire tampon vidéo pour déterminer les débits comme suit :

- Les représentations vidéo inférieures sont demandées lorsque le niveau du tampon.
- Les représentations vidéo supérieures lorsque le niveau du tampon est élevé.
- Le taux de remplissage de la mémoire tampon est contrôlée en sélectionnant la représentation vidéo selon la bande passante réseau calculée à partir du tampon en utilisant le ratio :

"Temps téléchargement segment/Taille segment".

— **Solution OSCAR (Optimized Stall-Cautious Adaptive bitRate) :**

Cette solution est proposée par Zahran et al.[189] pour assurer la diffusion vidéo à débit variable et pour éviter l'épuisement de la mémoire tampon tout en maintenant une haute qualité vidéo. Dans cette solution, la mémoire tampon est divisé en trois zones :

$$0 < \text{faible} < \alpha_1 < \text{transitoire} < \alpha_2 < \text{élevée}.$$

- Si l'état de remplissage de la mémoire tampon est faible, la représentation vidéo inférieure est demandée, afin d'accélérer le remplissage du tampon.
- Si l'état de remplissage de la mémoire tampon est élevée, le client cible la diffusion des représentations vidéo supérieures tout en évitant les périodes d'épuisement de la mémoire tampon.
- Au niveau transitoire, l'algorithme favorise la demande des représentations vidéo inférieures quand le taux de remplissage de la mémoire tampon diminue. Inversement, il maintient la même qualité lorsque le taux de remplissage de la mémoire tampon.

Le Tableau (5.7) présente les principales caractéristiques des solutions utilisées pour notre évaluation.

Solution et Acronyme	Politique pour la sélection du premier segment	Politique pour la gestion de la mémoire tampon	Sensible à la QoE	Performance en terme de quantité de donnée transférée	Modèle mathématique utilisé	Catégorie ABR
Apple HTTP Live Streaming (HLS)	70% du débit	1 niveau	Non	- -	Analyse statistique	TBA
Near-optimal bitrate adaptation for online videos (BOLA)	Selon le débit	1 niveau	Non	-	Optimisation de Lyapunov	TBA
Optimized Stall-Cautious Adaptive bitRate (OSCAR)	Selon le débit	3 niveaux	Non	-	Analyse statistique	BBA
Quality of Experience aware Adaptive BitRate (Q2ABR)	Proche de 50% du débit	3 niveaux	Oui	+	Apprentissage par renforcement	Hybride

TABLE 5.7 – Comparaison des méthodes utilisées pour l'évaluation

5.5.2 Évaluation

L'évaluation de notre solution "Q2ABR" est réalisée en suivant les trois étapes suivantes :

- Étape 1 : **Collecte des traces réelles :**

Afin de calculer la QoE des usagers, notre modèle d'estimation (Section (5.3.3)) utilise comme entrée la bande passante réseau, le taux de perte et la représentation vidéo. Comme première étape, nous avons réalisé une campagne de collecte en utilisant le programme "Python" ¹² et l'utilitaire "pings" ¹³. Les traces concernaient 4 métriques du réseau : la bande passante réseau (*Kbits/s*), la latence (*ms*), la gigue (*ms*) et le taux de perte (%). Chaque trace indique les conditions du réseau et sont mesurées sur des périodes d'observation qui s'étendent approximativement sur une période de 5 secondes. Ces traces sont recueillies dans différents modes de transport (bus, métro régional desservant Paris et sa région (RER) et train). Elles sont en accès libre sur le lien [27]. La Figure (5.9) présente les variations de la bande passante dans les traces récoltées.

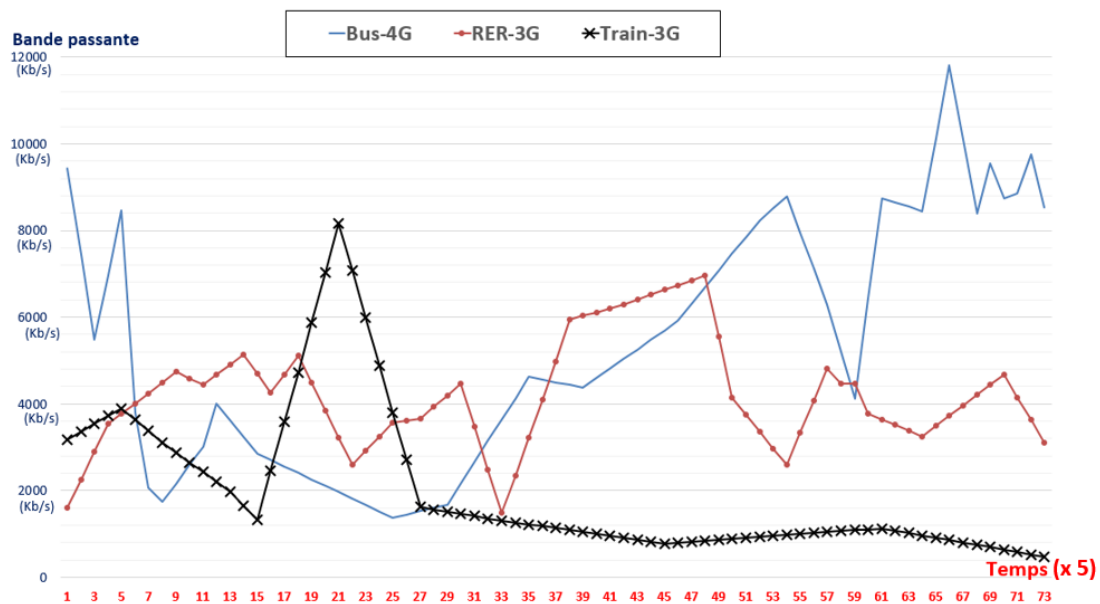


FIGURE 5.9 – Variations de la bande passante des traces réseaux collectées.

12. <https://pypi.org/project/pyspeedtest/>

13. <https://linux.die.net/man/8/ping>

Ces traces nous indiquent que chaque mode de transport a une distribution différente de la bande passante. Le trajet en bus en utilisant la technologie 4G est celui qui donne les meilleurs résultats par rapport aux autres, avec une valeur de bande passante minimale égale à 1.5 (Mb/s) et une valeur maximale de 11.7 (Mb/s). Les trajets en RER avec la technologie 3G, garantit une valeur minimale de la bande passante de 1.6 (Mb/s) et une valeur maximale de 7 (Mb/s). Le trajet en train avec la technologie 3G est celui ayant la bande passante la plus faible, où la valeur minimale est inférieurs à 800 (Kb/s) et la valeur maximale est égale à 8.1 (Mb/s). On peut donc constater sur ces traces une grande variabilité de la performance des réseaux.

- Étape 2 : **Développement d'un simulateur à évènements discrets :**

La performance de la solution "Q2ABR" de diffusion vidéo à débit variable est évaluée à l'aide du simulateur à évènements discrets comme proposé par l'étude [53]. Pour construire le simulateur, la boîte à outils *R* software [1] a été utilisée. Les méthodes BOLA[156] , HLS[115], OSCAR[189] ainsi que la solution "Q2ABR" (Section (5.4.3)) ont été implémentées dans le simulateur. Les conditions réseaux récoltées (étape 01) ainsi que la représentation vidéo sélectionnées par chaque méthode sont utilisées pour calculer les scores MOS des usagers. Dans le simulateur, nous avons considéré une taille de la mémoire tampon de 10 secondes et nous avons utilisé une séquence vidéo de durée égale à 60 secondes. Celle-ci est encodée avec des segments de 2 secondes en différentes représentations vidéo comme dans le Tableau (5.3).

- Étape 3 : **Évaluation**

Notre évaluation repose sur les métriques présentées dans le Tableau (5.8) qui sont reconnues comme des indicateurs clés pour évaluer les méthodes de diffusion à débit variable [183]. La première métrique est le nombre d'épuisements de la mémoire tampon (n_{st}). Elle indique le nombre de blocages de la séquence vidéo qui sont causés par l'épuisement de la mémoire tampon. La deuxième métrique concerne le nombre moyen de changements de qualité (n_{sw}). Celle-ci représente le nombre total de changements de qualités durant toute la durée de la séquence vidéo. La troisième métrique est la quantité totale de données transférées (S_d) depuis le serveur au client. La quatrième métrique (n_{mos}) est la moyenne des scores MOS calculés et,

tandis que la dernière métrique (sd_{mos}) est la variabilité des scores MOS calculés.

Sym	Description	Unité	Impact
n_{st}	Nombre moyen d'épuisement du tampon.	Entier	Négativement
n_{sw}	Nombre de changements de qualité.	Entier	Négativement
S_d	Total des données téléchargées.	Kbits/s	Négativement
n_{mos}	La moyenne des valeurs MOSs prédites.	/ 5	Positivement
sd_{mos}	L'écart type des valeurs MOSs prédites.	/ 5	Négativement

TABLE 5.8 – Métrique utilisées pour la comparaison

(i) Nombre moyen d'épuisement de la mémoire tampon (n_{st}) :

Le nombre moyen d'épuisement de la mémoire tampon résultant de l'utilisation des différentes méthodes ABRs a un impact direct sur la QoE des usagers comme démontré dans les études [44, 170]. En effet, si celui-ci est très élevé, il peut conduire à une insatisfaction de l'utilisateur ou un abandon du service. La Figure (5.10) présente les résultats de comparaisons des ABRs en termes de nombre moyen d'épuisement de la mémoire tampon.

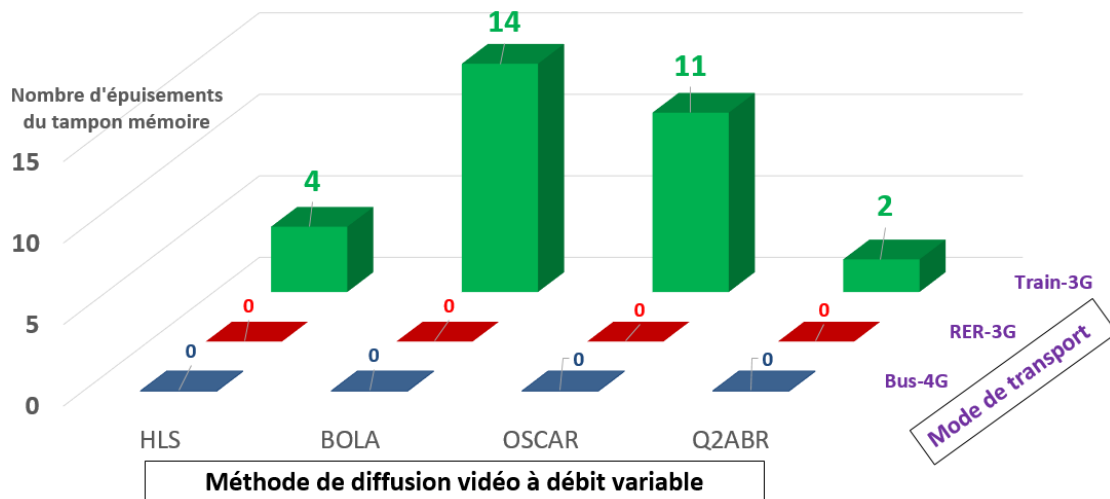


FIGURE 5.10 – Performances en termes de nombre d'épuisement de la mémoire tampon

Nous observons que notre proposition ($Q2ABR$) évite l'épuisement du tampon, lors des

trajets du bus et en RER. Dans le cas du trajet en train (réseau 3G), notre proposition assure le nombre de d'épuisement de la mémoire tampon le plus faible par rapport aux autres ABRs (HLS, BOLA et OSCAR) testés avec deux situations de tampon vide. Ce nombre non nul est justifié par les conditions réseau qui se sont détériorées d'une manière importante lors du trajet en train (c.f. Figure (5.9)).

(ii) Nombre de changements de qualité (n_{sw}) :

Selon l'étude [183], le nombre de changements de qualité (n_{sw}) est un paramètre important lors de la comparaison des méthodes de diffusion vidéo à débit variable. En effet, un bon algorithme d'adaptation du flux vidéo doit garantir un nombre minimum de changements de qualité. La Figure (5.11) présente le nombre de changements pour les différents ABRs lors de l'utilisation des différents mode de transports. Nous observons que le nombre de changements de qualité

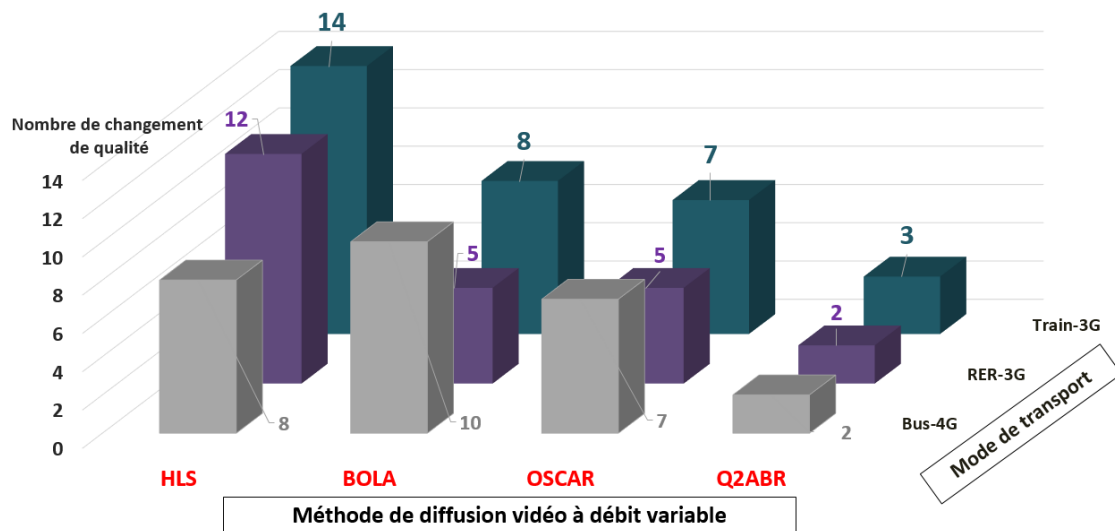


FIGURE 5.11 – Performances en termes de changement de qualité vidéo (n_{sw}) varie d'une méthode à une autre selon le mode de transport utilisé. En outre, nous constatons que notre solution "Q2ABR" réduit le nombre de changement de qualités : elle réalise, au plus, trois changements de qualités pour chaque type de trajet. Par exemple, lors de l'utilisation du train avec la technologie 3G (Figure (5.11)), notre proposition nécessite deux fois moins de changements que les méthodes ABRs testées (HLS (78%), BOLA (62%))

et OSCAR (57%). Les performances de notre proposition dans ce cas sont justifiées par le fonctionnement du module ("AgentAR") d'apprentissage par renforcement (Section 5.4.2) qui enclenche les actions (a_1 ou a_3) que quand c'est nécessaire (mémoire tampon vide ou grand changement du MOS). Sinon, on utilise la même qualité tant qu'on considère que l'utilisateur est satisfait ($MOS > 3$ d'après [2, 118]) et tant que la mémoire tampon maintient un bon taux de remplissage.

(iii) Total des données téléchargées (S_d) :

Le résultat du total des données téléchargées par chaque méthode est présenté dans la Figure (6.2). Cette métrique permet aux opérateurs de connaître le coût d'utilisation du réseau en quantifiant la quantité de données qui transite via leurs infrastructures.

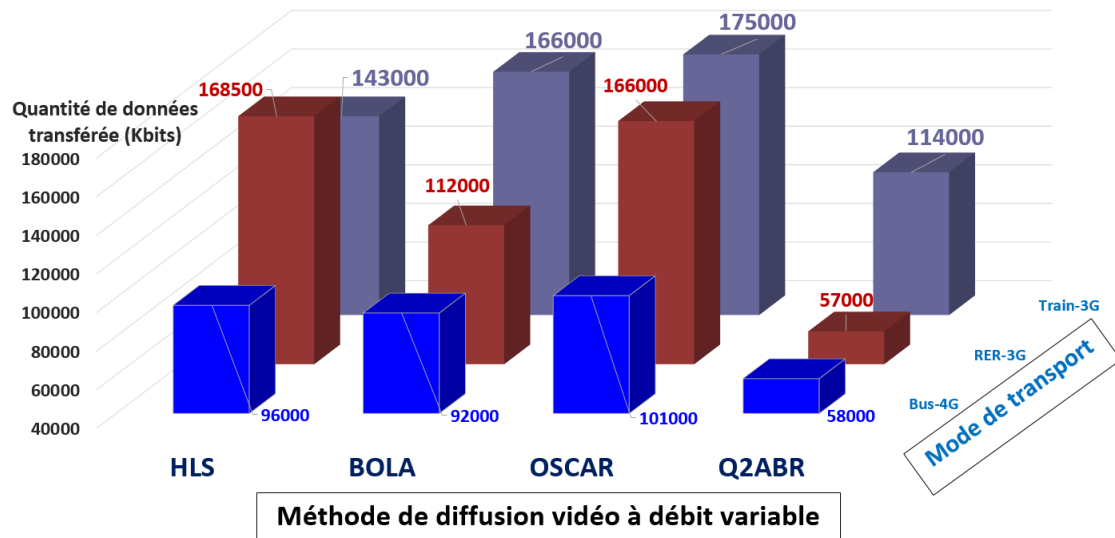


FIGURE 5.12 – Performances en termes du total des données téléchargé

Nous observons ici que notre proposition "Q2ABR" permet de réduire la quantité de données transférée par rapport aux autres méthodes étudiées (HLS, BOLA, et OSCAR) lors des différents modes de déplacement (Train, Bus et RER). En effet, elle réduit le trafic transféré de 37% à 42% lors des trajets en train (réseau 3G), et 20% à 35% dans un trajet en RER (réseau 3G). Par ailleurs, quand les conditions réseaux sont bonnes (lors du trajet en bus avec le réseau 4G), notre solution permet de réduire la quantité de données transférée de moitié par rapport

aux autres solutions testées (HLS (66%), BOLA (49%) et OSCAR (65%)). Ces résultats sont justifiés par l'utilisation d'une représentation vidéo initiale (qualité du premier segment) fixée en fonction de la bande passante calculée au lancement de la vidéo (Section 5.4.3). Comparée par rapport à la méthode HLS[115], notre méthode préconise une représentation vidéo inférieure pour le premier segment vidéo pour accélérer le remplissage de la mémoire tampon vidéo. Ensuite, notre approche essaye d'utiliser la même qualité tant que l'utilisateur est satisfait et tant que le tampon a un bon taux de remplissage. Ceci permet de réduire la quantité de données transférées ce qui permet de réduire la charge de l'utilisation des installations des opérateurs.

(iv) Satisfaction utilisateur (n_{mos} et sd_{mos}) :

Étant donné le caractère subjectif, la mesure du score MOS moyen et son écart type sont nécessaires pour connaître la satisfaction globale des usagers du service. En effet, plus la moyenne est grande et l'écart type est réduit, plus la satisfaction est meilleure. La Figure (5.13) présente les moyennes calculées des scores MOS prédits pour les différentes méthodes utilisées pour l'évaluation.

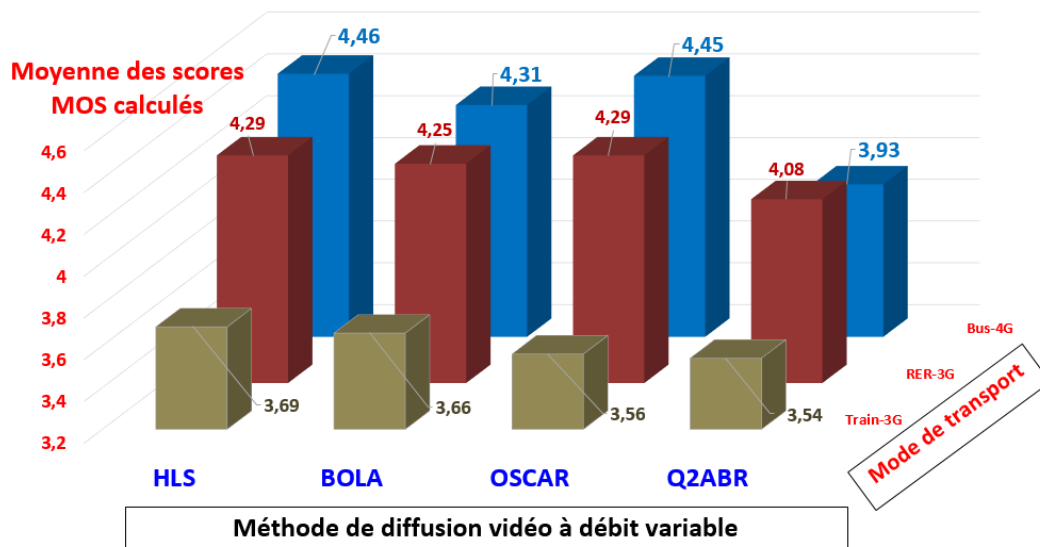


FIGURE 5.13 – Performances en termes de la moyenne des scores MOS prédite

Il en résulte que notre solution "Q2ABR" donne des résultats moins bons que ceux des

autres méthodes ABRs (HLS, BOLA et OSCAR). Ceci est justifié par le fait que ces méthodes favorisent la sélection des représentations vidéo supérieures qui permettent d'augmenter le score MOS moyen. Cela est vérifié dans la Figure (6.2), où les trois algorithmes testés (HLS, BOLA et OSCAR) donnent le plus grand taux de transfert de données en "Kbits" (S_d). Cependant, nous constatons que notre solution assure un bon niveau de QoE avec un MOS moyen (n_{mos}) supérieur à 3.5 sur une échelle de 5 [2, 118]).

L'autre résultat intéressant de notre proposition consiste dans la minimisation des variations des scores MOS calculés. Autrement dit, assurer une bonne qualité vidéo perçue par l'utilisateur en toutes circonstances en minimisant les variations de cette qualité (scores MOS calculés). En effet, nous avons constaté que la variabilité de la QoE est un facteur qui impacte le ressenti de l'utilisateur, d'où l'intérêt des résultats que nous présentons dans la Figure (5.14).

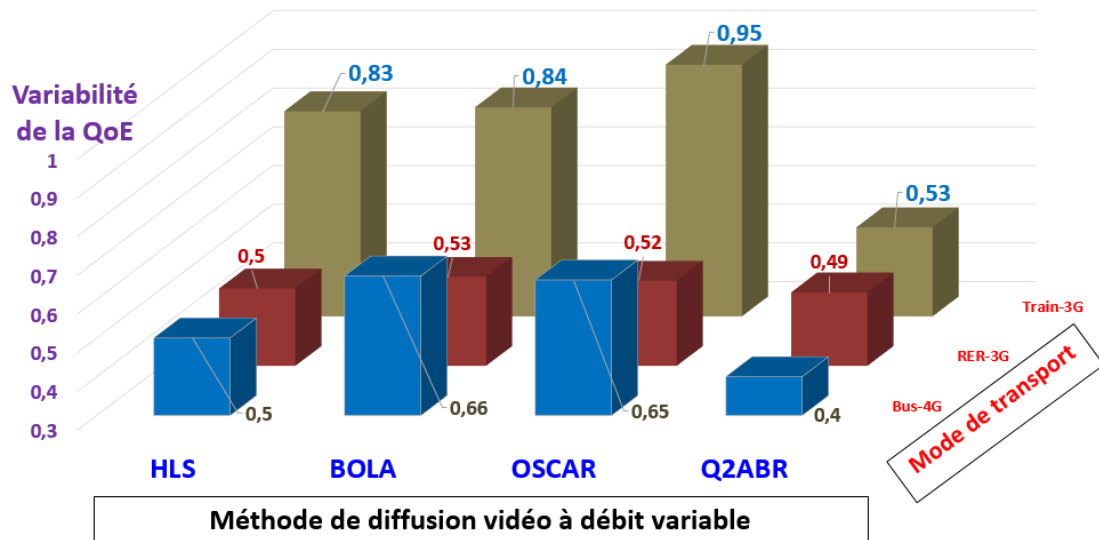


FIGURE 5.14 – Performances en termes des variations des scores MOS calculés

On peut observer que notre approche "Q2ABR" améliore de plus de 35% la valeur de variabilité de la QoE lors du trajet en train avec le réseau 3G (conditions réseau moins bonnes), et de plus 20% lors du trajet en bus avec le réseau 4G (conditions réseau plus favorables). Ces résultats sont justifiés par la politique de sélection utilisée par l'agent "AgentABR" (Figure (5.4)) dans laquelle la pénalité (l'équation (5.2)) est calculée selon les variations des scores MOS calculés.

Par conséquent, nous observons clairement que notre proposition offre de meilleurs résultats en termes de variation des scores MOS (sd_{mos}) par rapport aux autres approches.

5.5.3 Discussion

Nous avons développé une nouvelle approche, notée "Q2ABR", de diffusion vidéo à débit variable en utilisant l'apprentissage par renforcement pour la sélection des représentations vidéo choisies et le mécanisme de détection de point de changement pour le suivi de la satisfaction usager. L'objectif de notre proposition est de garantir une bonne qualité perçue par l'utilisateur tout en minimisant les variations de cette qualité.

Premièrement, nous avons construit un modèle qui prédit les scores MOS des utilisateurs de la vidéo en utilisant les méthodes d'apprentissage automatique (*CBF*). Deuxièmement, nous avons collecté des traces réelles de trois scénarii de déplacements (trajet en bus, en RER et en train). Ensuite, nous avons proposé une nouvelle solution de diffusion de vidéo à débits variables.

Pour valider notre solution, nous l'avons comparé à trois solutions existantes dans la littérature : HLS [115], BOLA [156] et OSCAR [189]. Un ensemble de métriques, présentées dans le Tableau (5.8), est utilisé à des fins de comparaison.

D'après les résultats, nous constatons que notre approche ("Q2ABR") évite le problème d'épuisements de tampon en le subdivisant en trois parties (faible, transitoire et stable (élevé)) et en surveillant les changements des valeurs calculées de la QoE en termes de scores MOS.

La politique mise en œuvre pour la sélection des représentations vidéo des segments consiste à offrir une QoE suffisante et satisfaisante couplée à l'utilisation d'une approche de détection de points de changement de QoE. Cette solution nous a permis de réduire l'utilisation de la bande passante comparativement aux solutions existantes.

Enfin, alors que la plupart des chercheurs travaillant sur la diffusion vidéo à débit variable abordent la question de l'étude et de l'optimisation de la représentation vidéo sélectionnée et de la QoE usager, l'idée maîtresse défendue ici est d'assurer la satisfaction de l'utilisateur en proposant une bonne qualité, qui peut ne pas être la plus élevée possible, tout au long de la séquence vidéo. Donc, à la place de maximiser le ressenti instantané de l'utilisateur (score MOS), notre solution assure une variabilité minimale de ce ressenti à travers le temps.

5.6 Conclusion

Dans ce chapitre, nous avons étudié le problème de diffusion vidéo à débit variable utilisant la norme Dash. Une méthode basée sur l'apprentissage par renforcement a été proposée. Cette méthode modifie de manière adaptative la représentation vidéo demandée pour assurer une bonne qualité d'expérience (QoE) usager et optimiser l'utilisation de la bande passante. Le modèle de prédiction présenté dans le chapitre précédent (CBF) a été utilisé pour prédire la satisfaction des usagers en termes du MOS en utilisant la bande passante, la représentation vidéo et le taux de perte comme paramètres d'entrée. Les résultats expérimentaux avec une taille de mémoire tampon de 10 secondes ont montré que la méthode proposée peut assurer un minimum de variations de la QoE usager tout en fournissant une aussi bonne QoE moyenne que les méthodes existantes dans la littérature (BOLA, HLS et OSCAR).

Chapitre 6

Conclusion générale et perspectives

Au terme de ce manuscrit, nous nous proposons de faire un récapitulatif des travaux réalisés, d'analyser globalement les résultats obtenus et de proposer une feuille de route en dressant des perspectives pour de nouveaux axes de recherche.

6.1 Travail réalisé et contributions

Les acteurs de l'Internet, notamment les fournisseurs de service et les opérateurs réseaux, utilisent des stratégies de supervision réseau pour assurer une bonne qualité de service (QoS) de bout-en-bout à leurs usagers. Cependant, il s'est avéré que les méthodes basées uniquement sur la QoS peuvent être insuffisantes, voire inappropriés dans certaines situations. Partant de ce constat, les acteurs de l'Internet ont changé leurs stratégies en intégrant l'utilisateur dans le contrôle du réseau. Cela est rendu possible en calculant en amont la qualité réelle que devait percevoir l'utilisateur final, plus connue sous le nom "expérience de l'utilisateur (QoE)". Le concept de la QoE représente un nouveau paradigme dont l'objectif principal est de permettre à ces acteurs de garantir une meilleure satisfaction des usagers. Cette thèse s'inscrit dans ce contexte.

Pour commencer, le manuscrit de thèse donne une présentation générale de l'état de l'art des méthodes utilisées pour calculer la QoE perçue par les usagers. Ces méthodes peuvent être classées en deux types selon que les humains interviennent ou non dans l'évaluation de la QoE usager. Celle-ci est dite subjective si les utilisateurs y participent en fournissant leurs opinions. Elle est dite objective si l'évaluation est faite automatiquement sans intervention humaine. Ceci est possible soit via l'analyse mathématique de la relation QoE et les facteurs qui l'impactent ou soit via une approche pilotée par les données et des méthodes d'apprentissage automatique.

Afin de construire notre propre modèle de calcul de la QoE usager, appelé en anglais *Combined Belief Functions (CBF)*, nous avons, dans un premier temps, conduit une campagne de test, en utilisant la méthode subjective "Note Absolue par Catégorie", connue sous l'appellation anglaise : *Absolute Category Rating (ACR)* [2]. Notre but est de récolter une base de données contenant les différents facteurs susceptibles d'impacter la QoE usager [93]. Ces facteurs concernent le service vidéo et les retours des usagers en termes de "note d'opinion moyenne", connue sous l'appellation anglaise : *Mean Opinion Score (MOS)* [137]. Dans un deuxième temps, nous avons proposé une approche pour la sélection des facteurs qui nous permet d'optimiser globalement le calcul de la QoE perçue par l'utilisateur. Cette approche a l'avantage de sélectionner les facteurs les plus pertinents. Dans un troisième temps, nous avons introduit notre modèle "CBF" qui calcule la QoE en terme du score MOS. Ce modèle se base sur l'usage des fonctions de croyance et de la fusion d'information en utilisant le modèle des Croyances Transférables (MCT) [131]. L'évaluation réalisée, grâce à la base de données récoltée, a permis de vérifier que le modèle proposé améliore les performances de calcul en terme d'erreur quadratique et d'erreur absolue moyenne ainsi qu'en terme de taux de corrélation.

Pour tester les performances du modèle d'estimation proposé dans un cas d'usage concret, nous avons mis en œuvre une solution globale de streaming vidéo à débit variable (Adaptive Bit Rate (ABR)) compatible à la norme Dash [66]. Cette solution est appelée "Q2ABR" pour "*Quality of Experience aware Adaptive Bit Rate*". Elle se base sur les mesures de la QoE calculées ainsi que sur un modèle paramétrique qui optimise en temps réel la qualité de la

vidéo en fonction de la variation des mesures instantanées de la QoE calculées. L'objectif de cette solution est d'améliorer l'expérience globale de l'utilisateur, tout en optimisant l'utilisation du réseau en ajustant en temps réel la représentation des segments vidéo à afficher. À des fins de validation, la solution "Q2ABR" proposée a été comparée à 3 méthodes existantes dans la littérature en utilisant une taille de mémoire tampon de seulement 10 secondes. Les résultats obtenus dans cette étude ont montré l'efficacité de notre solution en termes de nombre d'épuisement de la mémoire tampon, de nombre de changements de qualité et des variations de la qualité perçue par l'utilisateur. Par ailleurs, cette solution permet aussi de réduire la quantité de données transférée, depuis le serveur au client, par rapport aux autres méthodes étudiées.

6.2 Enseignement et retours d'expérience

Ces dernières années ont été le théâtre d'une évolution dans la gestion des réseaux. En effet, d'une vue dite "centrée réseau", nous sommes aujourd'hui passés à une vue dite "centrée utilisateur". Ce paradigme consiste à contrôler le réseau en utilisant une supervision basée sur la QoE usager. Néanmoins, la question qui se pose est de quelle manière mesure-t-on cette QoE ? Deux écoles existent : celle qui plaide pour une évaluation subjective en demandant explicitement à l'utilisateur son ressenti et celle qui plaide pour une évaluation objective et non intrusive en utilisant les outils de l'apprentissage automatique.

À l'issue de ce travail, nous pourrions dégager plusieurs renseignements que nous résumons ci-dessous :

- Le calcul de la QoE basé sur les méthodes d'apprentissage automatique permet d'obtenir de meilleurs résultats que celui qui utilise des modèles mathématiques classiques (IQX Hypothesis[143], régression linéaire[76], etc.). Le principal inconvénient de ces méthodes réside dans le fait que le modèle généré est spécifique aux données utilisées lors de la phase d'apprentissage. À cet égard, il convient d'apprendre à partir de toutes les configurations de données représentatives pour améliorer la capacité de généralisation des modèles proposés.

- La collecte des données issues de campagnes de test subjectives est une tâche complexe. Il est important que les chercheurs s'orientent vers la publication de leurs données afin de favoriser la réutilisation et la reproductibilité.
- L'utilisation des méthodes d'apprentissage automatique nécessite une étape de sélection d'un sous-ensemble de caractéristiques pertinentes [64, 79]. Il y a plusieurs façons pour réaliser cette sélection : soit par des approches de filtrage, soit par des approches enveloppantes (*wrapper*) ou encore par des approches intégrées (*Embedded*)[49]. Les approches enveloppantes (*wrapper*) sont aujourd'hui les plus communément utilisées. Néanmoins, des approches hybrides peuvent fournir de meilleurs résultats dans des contextes particuliers.
- L'utilisation du chiffrement au niveau applicatif, permet d'assurer plus de confidentialité aux usagers. Néanmoins, il pénalise les opérateurs réseau car ils ne disposent d'aucune information sur le flux de données transitant. Ainsi, il nous paraît indispensable que les opérateurs et les fournisseurs collaborent afin de garantir un meilleur service pour les usagers. Par exemple, le protocole *QUIC* [100] implémente cela à travers le champ *PLUS* (*previous SPUD (Session Protocol for User Datagrams)*) qui permet aux fournisseurs de services de partager certains indicateurs de performance aux opérateurs réseau. Néanmoins, il reste quelques préoccupations concernant les stratégies de partage de ressources réseaux entre les fournisseurs.

6.3 Perspectives et travaux à venir

Les perspectives d'évolution s'inscrivent dans la continuité du travail déjà accompli. Il s'agit, en premier lieu, d'améliorer la campagne de test pour construire une base de données de meilleure qualité. Dans cette perspective, le but est de faire face aux situations contradictoires, lors des tests, dans lesquelles un grand écart est constaté entre les valeurs de MOS données par les testeurs pour une combinaison de facteur donnée (même conditions de test : paramètres réseaux, caractéristiques du dispositif d'affichages, paramètres applicatifs). En fait, au lieu

d'utiliser des combinaisons aléatoires dans les tests, l'idée est de favoriser l'utilisation des combinaisons de facteurs qui mènent à des situations contradictoires. Autrement dit, utiliser les valeurs de facteurs qui détériorent les performances de calcul de la QoE à la place des valeurs donnant des valeurs calculées prévisibles. En effet, l'une des pistes prometteuses à explorer consiste à l'utilisation de l'apprentissage active (*Active Learning*) pour la réalisation de campagne de test.

De plus, les progrès technologiques récents ont permis une prolifération constante de nouveaux services vidéo (immersifs et interactifs) qui posent des exigences croissantes aux opérateurs comme les jeux mobiles en ligne, la vidéo UHD (4K/8K), les environnements immersifs, la vidéo en 3D et la réalité augmentée. Une deuxième perspective de ces travaux s'inscrit dans cette optique, où nous pensons à procéder à la diversification de l'utilisation du modèle proposé de calcul de la QoE pour ces services émergents. En effet, une étude de la faisabilité de la campagne d'évaluation pour le contexte de chaque service vidéo apparaît alors comme nécessaire, où des campagnes de tests bien spécifiques à chaque service vidéo doivent être réalisées (terminaux adaptés, contexte adéquat, ...etc.) afin de construire la base de données nécessaire et de mettre en service notre modèle de calcul CBF.

Comme troisième perspective, nous pensons à travailler sur un modèle générique d'estimation de la QoE qui permettra d'évaluer différents services autres que le service vidéo. L'idée est d'améliorer la généricité du modèle en y intégrant plus de paramètres applicatifs (QoA) spécifiques à chaque service (VoIP, navigation sur le web, conférence audio, ...etc.). Il importe toutefois de souligner que, même si les résultats de nos travaux ont abouti à la proposition d'un modèle générique de base pouvant s'adapter à plusieurs profils usagers, il n'empêche qu'il peut être moins performant pour certains utilisateurs. Une piste d'amélioration consisterait à proposer à partir d'un modèle générique, un modèle personnalisé de calcul de la QoE pour chaque utilisateur. Ceci en intégrant de nouveaux exemples et de facteurs spécifiques dans la construction de ce modèle. L'idée maîtresse consisterait à enrichir la base de données, utilisée initialement pour construire le modèle générique, par de nouveaux exemples

spécifique à chaque utilisateur. Des nouveaux facteurs peuvent être également rajoutés pour affiner le modèle personnalisé comme les paramètres affectifs (facteurs émotionnels). Ceux-ci représentent les expressions faciales du visage comme le dégoût, la surprise, la joie, la tristesse, etc. En effet, l'intégration de ces facteurs permettra de raffiner et de personnaliser le modèle d'estimation en adaptant le service selon les émotions instantanées de l'utilisateur de service. Pour arriver à cette fin, on peut utiliser plusieurs types de capteurs comme la caméra.

Comme autre perspective, nous pensons à travailler sur un service à valeur ajoutée pour les opérateurs avec différenciation par profil usager. En effet, nous envisageons l'intégration du profil de l'utilisateur (prix, expérience utilisation vidéo, besoins, temps utilisation de service) dans notre modèle de calcul de QoE (CBF). Cette intégration permettra ainsi à l'opérateur de fournir un bien meilleur service à ses utilisateurs au juste prix, en les classifiant selon les différents profils définis auparavant. Nous pensons que la vision de l'administration du réseau évoluera vers une méta-administration où l'opérateur n'aura qu'à définir des objectifs de haut niveau fixés sur la base des profils usagers prédéfinis. Dans ce réseau, l'administrateur aura la possibilité d'agir sur les mécanismes de gestion et de contrôle de l'ensemble des équipements du réseau de l'utilisateur final (du serveur d'applications jusqu'à la borne d'accès au réseau) .

Pour conclure, l'usage de la QoE dans le contrôle du réseau sera de plus en plus utilisé. La tendance actuelle est qu'il serait intéressant de travailler sur des approches adaptatives régulées par retours d'expérience (QoE). Une continuité à ce travail serait ainsi de diversifier l'utilisation de notre modèle de calcul CBF pour d'autres cas d'usage dans le contrôle intelligent dans le cadre de mécanismes clés, comme par exemple : contrôle bande passante en temps réel dans les villes intelligentes (Smart Cities), routage intelligent à travers l'Internet, transcoding en ligne de la vidéo au niveau des CDNs ou les serveurs clouds, compensation des retards dans les réseaux Ad-Hoc de véhicules (VANet), Optimisation du trafic de la vidéo Ultra Haute Définition via SDN, etc. L'utilisation de la QoE apparaît alors comme un élément absolument indispensable de l'ensemble du processus de développement de nouvelles approches adaptatives pour la gestion du réseau dans le futur.

Annexe 1 : Base de données QoE de l'environnement contrôlé

La base de données présentée dans le chapitre 2 et disponible dans l'URL (<https://github.com/PhDAmour/QoEcontrlledLaboratory>), est pour laquelle quelques détails sont donnés dans cette partie. Cette base de données est recueillie à l'aide de la méthode subjective du ACR (Absolute Category Rate)[2] basée sur les notes d'opinion moyenne (Mean Opinion Score (MOS)). Elle fournit un panel représentatif des différents facteurs d'influence de la QoE, qui permet à la communauté des chercheurs de travailler afin d'étudier la QoE des usagers du service vidéo.

Pour construire cette base de données, une campagne de test est réalisée dans le laboratoire LiSSi (<http://lab.lissi.fr/>) dans la ville de Vitry sur Seine. 62 testeurs ont participé à la campagne de test. Tous étaient des chercheurs et des étudiants de différentes disciplines âgés de 19 à 41 ans avec peu ou pas d'expérience dans ce genre d'expérimentation. Les facteurs collectés concernent de nombreux paramètres vidéo que le lecteur VLC fournit comme indicateur.

1- Source :

1.1. Créateurs de la base de données :

-> Lamine Amour (lamine.amour@u-pec.fr)

-> Sami Souihi (sami.souihi@u-pec.fr)

-> Abdelhamid Mellouk (mellouk@u-pec.fr)

1.2. Télé-chargement : Trois formats pour le téléchargement sont disponibles :

-> name.csv : Microsoft Excel 2007 file.

-> name.arff : Weka files

-> name.data : Open Document format

URL du téléchargement : <https://github.com/PhDAmour/QoEcontrlledLaboratory>.

2- Informations sur la base de données :

L'ensemble de données a été construit à partir d'une campagne de test réalisée dans un environnement contrôlé, où 560 échantillons couvrant 19 facteurs de la QoE ont été récoltés. Les vidéos utilisées sont de différents types/complexités comme nous l'illustre la Figure (6.1).

ID	Content	Size [Pixel]	Frame rate [frame/second]	Size [Megaoctets]	Complexity	Complexity class	Normalized Bit rate
1	New 1	1920x1080	30	97	5,21	0	5,2
2	New 2	1920x1080	30	90	5,913	0	4,73
3	New 3	1920x1080	30	101	5,333	0	5,41
4	Sports 1	1920x1080	30	329	9,01	3	17,37
5	Sports 2	1920x1080	30	343	9,298	3	18,99
6	Sports 3	1920x1080	30	186	7,23	2	9,88
7	Documentary 1	1920x1080	30	102	5,32	0	5,39
8	Documentary 2	1920x1080	30	241	8,01	3	12,65
9	Documentary 3	1920x1080	30	182	7,12	2	9,53
10	Music 1	1920x1080	30	232	7,89	2	12,17
11	Music 2	1920x1080	30	163	0,495	0	0,85
12	Music 3	1920x1080	30	164	6,81	1	8,63
13	Cartoon 1	1920x1080	30	99	5,223	0	5,22
14	Cartoon 2	1920x1080	30	148	6,488	1	7,8
15	Cartoon 3	1920x1080	30	174	7,008	2	9,19
16	Lesson 1	1920x1080	30	10	0,164	0	0,55
17	Lesson 2	1920x1080	30	37	2,095	0	1,94
18	Lesson 3	1920x1080	30	41	2,465	0	2,18
19	Movie 1	1920x1080	30	103	5,333	0	5,41
20	Movie 2	1920x1080	30	166	6,855	1	8,76
21	Movie 3	1920x1080	30	122	5,874	0	6,42
22	Game 1	1920x1080	30	229	7,911	2	12,24
23	Game 2	1920x1080	30	190	7,342	2	10,22
24	Game 3	1920x1080	30	248	8,65	3	12,85

FIGURE 6.1 – Détails des vidéos utilisées.

L'emplacement géographique de la campagne de test était le laboratoire LiSSi (<http://lab.lissi.fr/>) autour de la ville de Paris en France. 62 testeurs ont participé à la campagne de test. Tous étaient des chercheurs et des étudiants de différentes disciplines âgés de 19 à 41 ans avec peu ou pas d'expérience dans l'expérimentation de l'évaluation vidéo.

3- Attributs de la base de données :

la base de données contient 18 facteurs et les scores MOS des participants aux tests. Chaque facteur appartient à une catégorie de facteurs QoE (Figure 1.2) :

- Paramètres du contenu de la vidéo d'origine (2 à 6)

ID	Facteur	Description	Valeurs
2	V_id-video	Identifiant de la vidéo.	1, 2, 3, ..., 24
3	V_content	Le contenu de la vidéo.	1 :Journal, 2 :Sports, ..., 8 :Jeux
4	V_norm-bitrate	La distribution du débit binaire vidéo.	Réel
5	V_complexity	La complexité de la vidéo.	Réel
6	V_complexity-class	La classe de complexité de la vidéo.	0,1,2 ou 3

- Indicateurs lecteur client vidéo VLC (7 à 14)

ID	Facteur	Description	Valeurs
7	QoA_VLCresolution	La qualité vidéo vidéo.	144p/240p/360p/480p/720p/1080p
8	QoA_VLCcaching	Valeur du cache du lecteur vidéo VLC.	[100, 2000] ms
9	QoA_VLCbitrate	Débit binaire vidéo moyen.	Réel
10	QoA_VLCframerate	Nombre moyen d'images par seconde.	[0,30] images par seconde.
11	QoA_VLCdropped	Nombre moyen d'images perdues par seconde.	[0,30] images par seconde
12	QoA_VLCaudiorate	Débit binaire audio moyen.	Réel
13	QoA_VLCaudioloss	Nombre moyen de paquets audio perdus par seconde.	Réel
14	QoA_BUFFERINGtime	Temps moyen de bufferisation (lancement).	[100, 10000] ms

- Caractéristiques du dispositif d'affichage (15)

ID	Facteur	Description	Valeurs
15	QoD_screen_size	Taille de l'écran utilisé lors des tests	1366x768, 1920x1200

- Profil de l'utilisateur (16 à 18)

ID	Facteur	Description	Valeurs
16	id_user	Identifiant du participant au test.	1, 2, 3,..., 62
17	QoU_age	Âge du participant au test.	19, 20, 21,..., 41
18	QoU_sex	Genre du participant au test	Homme ou Femme

- Score MOS de l'utilisateur (19)

ID	Facteur	Description	Valeurs
19	MOS	Note moyenne de la qualité de la vidéo	1, 2, 3, 4 ou 5.

- Nombre d'instances des cores MOS :

-> classe 1 (MOS = 1) : 138

-> classe 2 (MOS = 2) : 100

-> classe 3 (MOS = 3) : 103

-> classe 4 (MOS = 4) : 132

-> classe 5 (MOS = 5) : 89

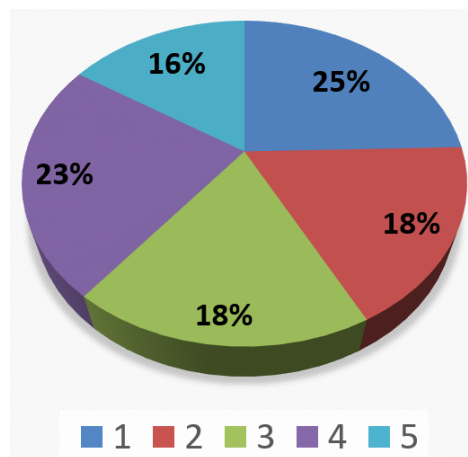


FIGURE 6.2 – Distribution des scores MOS récoltés

4- Études utilisant la base de données

(a) L. Amour, S. Souihi, S. Hoceini, A. Mellouk : An Open Source Platform for Perceived Video Quality Evaluation (Q2SWinet) : pages 139-140. November 2015.

(b) L. Amour, S. Souihi, S. Hoceini, A. Mellouk : A Hierarchical Classification Model of QoE Influence Factors. WWIC 2015 : pages 225-238, May 2015.

(c) Y. B. Youssef, A. Mellouk, M. Afif and S. Tabbane : Video Quality Assessment Based on Statistical Selection Approach for QoE Factors Dependency," 2016 IEEE Global Communications Conference (GLOBECOM), pages 1-6, December 2016.

Annexe 2 : Document d'instructions de la campagne test

Le fichier présenté dans cette partie est le document utilisé pour expliquer et former les participants aux campagnes de test subjective. Ce document est inspiré des travaux du projet de standardisation projet AVHD (<https://www.its.bldrdoc.gov/vqeg/projects/audiovisual-hd.aspx>) du groupe de travail VQEG (Video Quality Experts Group). Le document est présenté ci-dessous. À noter qu'il est signé à la fin par le participant ainsi que l'expérimentateur.

La campagne de test est réalisée dans le laboratoire LiSSi (<http://lab.lissi.fr/>) et les participants sont les étudiants et les chercheurs de ce laboratoire ainsi que la participation des personnes externes. La Figure (6.3) présente un aperçu de l'environnement de test réalisé.



FIGURE 6.3 – Description de l'environnement contrôlé de l'évaluation de la QoE

Introduction :

Merci d'être venus aujourd'hui pour participer à notre étude d'évaluation de la qualité d'expérience (QoE) usager du service vidéo. L'étude porte sur la qualité des séquences vidéo. Elle est menée par l'équipe de recherche Contrôle Intelligent des Réseaux (CIR) du laboratoire LiSSi (Université Paris Est Créteil). Le but de cette étude est de construire une base de données qui permet la création des systèmes de diffusion audio-vidéo. Cette étude s'intéresse à ce qui semble bon pour vous ainsi que pour l'utilisateur potentiel des appareils de la prochaine génération.

Objectif global :

Nous allons vous demander de regarder un certain nombre de séquences pour juger de la "qualité" de chacune d'entre elles. Nous expliquerons plus loin ce que nous entendons par "qualité". Ces séquences ont été traitées par différents systèmes, de sorte qu'elles peuvent ou non vous sembler différentes. Nous vous demanderons d'évaluer la qualité de chacun d'entre elles après l'avoir vu.

Étapes du test :

- Pour commencer l'étude, veuillez vous asseoir sur la chaise rouge devant la table préparée (Figure (6.3)).
- Les vidéos seront affichées sur le l'écran d'affichage (ordinateur, TV).
- Veuillez garder votre tête raisonnablement près de la position du milieu de l'écran d'affichage. C'est parce que les vidéos peuvent sembler un peu différentes d'une position à l'autre, et nous aimerions que tout le monde juge les vidéos à peu près de la même position.

Déroulement du test :

- Chaque vidéo sera d'une durée de de 30 secondes. Vous devrez appuyer sur le bouton "Commencer" (l'écran à votre gauche (Figure (6.3)) pour démarrer la lecture vidéo. Ensuite, la séquence vidéo sera affichée sur l'écran d'affichage qui se trouve en face de vous.
- Si la vidéo est terminée, vous devez ensuite voter en utilisant le deuxième écran à gauche (Figure (6.3)) en choisissant la qualité appropriée.
- Ce processus sera répété jusqu'à ce que vous ayez vu au moins 3 vidéos. Ensuite, vous pourriez arrêter le test quand vous voulez.

Questions :

Si vous avez des questions, veuillez vous adresser au responsable de l'expérience. Consentement du sujet : L'expérience du projet AVHD est menée au laboratoire LiSSi. Le but, la procédure et ainsi que les risques de la participation à l'expérience du projet AVHD m'ont été expliqués. J'accepte volontairement de participer à cette expérience. Je comprends que je peux poser des questions et que j'ai le droit de me retirer de l'expérience à tout moment. Je comprends également que le laboratoire LiSSi peut m'exclure de l'expérience à tout moment. Je comprends que toute donnée que je contribue à cette expérience ne sera pas identifiée avec moi personnellement, mais ne sera rapportée qu'en tant que moyenne statistique.

Signature du participant :

Signature de l'expérimentateur :

Bibliographie

- [1] R software, Institute for Statistics and Mathematics of WU, (Wirtschafts universität Wien). <http://www.r-project.org/>.
- [2] Union Internationale des Télécommunications (UIT-T) Recommandation *P800* : Méthodes d'Évaluation Subjectives de la Qualité de Transmission. pages 1–39, Mars 1998.
- [3] Union Internationale des Télécommunications (UIT-R) Recommandation *P911* : Méthodes d'Évaluation Subjective de la Qualité Audiovisuelle pour Applications Multimédias. pages 1–29, Avril 1999.
- [4] Union Internationale des Télécommunications (UIT-R) Recommandation *BT.500 – 1* : Méthodologie d'Évaluation Subjective de la Qualité des Images de Télévision. pages 1–48, Octobre 2002.
- [5] Union Internationale des Télécommunications (UIT-T) Recommandation *J.144* : Objective Perceptual Video Quality Measurement Techniques for Digital Cable Television in the Presence of a Full Reference. pages 1–156, Mars 2003, Access : <http://www.itu-t.org>.
- [6] Video Quality Experts Group, Multimedia Group Test Plan, Draft version 1.8. www.vqeq.org, December 2005.
- [7] Union Internationale des Télécommunications (UIT-R) Recommandation *P910* : Méthodes Subjectives d'Évaluation de la Qualité Vidéographique pour les Applications Multimédias. pages 1–42, Avril 2008.

- [8] Union Internationale des Télécommunications (UIT-T) Recommandation E.800 : Définition de termes Relatifs à la Qualité de Service. pages 1–32, 2008, Access : <http://www.itu-t.org>.
- [9] Caractéristiques du Codec SILK : . Accès : SkypeDeveloper, <http://developer.skype.com/silk>.
- [10] International Telecommunication Union - Telecommunication (ITU-T)-T Recommendation P.10/G.100 : Definition of Quality of Experience (QoE). pages 1-8. January 2006.
- [11] A. Arauzo-Azofra, J. Manuel Benitezand and J. L. Castro. Consistency Measures for Feature Selection. *Journal of Intelligent Information Systems, Volume 30, Issue 3*, Pages 273-292, June 2008.
- [12] H. A. Tran, A. Mellouk and S. Hoceini. *Media Networks Architectures, Applications, and Standards* . Media Networks Architectures, Applications, and Standards, CRC Press, pages 255–274, May 2014.
- [13] W.J Priyadarshana. and G. Sofronov. Breakpoint : An R Package for Multiple Break-Point Detection via the Cross-Entropy Method. *CRAN Repository* , pages 1–25, January, 2016.
- [14] A. Appriou. Multisensor Signal Processing in the Framework of the Theory of Evidence. *In Application of Mathematical Signal Processing Techniques to Mission Systems, Research and Technology Organization*, pages 1–31, November 1999.
- [15] A. Aziza, M. Oubrich and K. Solberg Sóilen. The impact of CRM on QoE : An Exploratory Study from Mobile Phone Industry in Morocco. *Journal of Intelligence Studies in Business (JISIB)*, 5(2) : pages 22–35, April 2015.
- [16] A. Balachandran, V. Sekar, A. Akella, S. Seshan, I. Stoica and H. Zhang. Developing a Predictive Model of Quality of Experience for Internet Video. *SIGCOMM Comput. Commun. Rev.*, 43(4) : pages 339–350, August 2013.
- [17] A. Ghosh, V. Aggarwal and F. Qian. A Rate Adaptation Algorithm for Tile-based 360-degree Video Streaming. *Computing Research Repository (CoRR) Journal*, abs/1704.08215 : pages 1–13, April 2017.

- [18] A. Khan, L. Sun and E. Ifeachor. Content Clustering Based Video Quality Prediction Model for MPEG4 Video Streaming over Wireless Networks. pages 1–5, June 2009.
- [19] A. Khan, L. Sun and E. Ifeachor. Learning Models for Video Quality Prediction over Wireless Local Area Network and Universal Mobile Telecommunication System Networks. *IET Communications*, 4(12) : pages 1389–1403, August 2010.
- [20] A. Khan, L. Sun and E. Ifeachor. An ANFIS-Based Hybrid Video Quality Prediction Model for Video Streaming over Wireless Networks. In *The Second International Conference on Next Generation Mobile Applications, Services, and Technologies*, NGMAST '08, pages 357–362, September 2008.
- [21] A. Martin, A. L. Jusselme and C. Osswald. Conflict measure for the Discounting Operation on Belief Functions. In *11th International Conference on Information Fusion*, pages 1–8, June 2008.
- [22] A. Mellouk, S. Hoceini and H. A. Tran. Quality of Experience vs. Quality of Service : Application for a CDN Architecture. In *2013 21st International Conference on Software, Telecommunications and Computer Networks - (SoftCOM 2013)*, pages 1–8, September 2013.
- [23] A. Mellouk, S. Hoceini and H. A. Tran. *Quality of Experience Models for Multimedia Streaming*. ISTE Ltd and John Wiley Sons Inc, United Kingdom, pages 1-135, February 2014.
- [24] A. Aaron, Z. Li, M. Manohara, J. Y. Lin, E. C. H. Wu, and C. C. J. Kuo. Challenges in cloud based ingest and encoding for high quality streaming media. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 1732-1736. September 2015.
- [25] E. Alfaro, M. Gamez, and N. García. adabag : An R Package for Classification with Boosting and Bagging. *Journal of Statistical Software, Articles*, 54(2) : pages 1–35, September 2013.
- [26] M. Alreshoodi and John Woods. Survey on QoE\QoS Correlation Models For Multimedia Services. *Computing Research Repository (CoRR) Journal*, Volume 1306.0221 :53–72, June 2013.

- [27] L. Amour. Network QoS parameters traces. <http://tincnet.fr/tracesQoSstoABRs>, January 2018.
- [28] L. Amour, S. Souihi, S. Hoceini, and A. Mellouk. Platform Video :. <https://youtu.be/pMvfVQYp1Vk>, 2015.
- [29] L. Amour, S. Souihi, S. Hoceini, and A. Mellouk. A Hierarchical Classification Model of QoE Influence Factors. *13th International Conference on Wired and Wireless Internet Communications*, pages 225 – 238. May 2015.
- [30] R. Aparicio-Pardo, K. Pires, A. Blanc, and G. Simon. Transcoding Live Adaptive Video Streams at a Massive Scale in the Cloud. In *Proceedings of the 6th ACM Multimedia Systems Conference, MMSys '15*, pages 49-60, March 2015.
- [31] S. Aroussi and A. Mellouk. Survey on Machine Learning-based QoE-QoS Correlation Models. In *International Conference on Computing, Management and Telecommunications (ComManTel)*, pages 200–204, April 2014.
- [32] B. T. Abe, O. O. Olugbara and T. Marwala. Hyperspectral Image Classification using Random Forests and Neural Networks. *Proceedings of the World Congress on Engineering and Computer Science (WCECS), Volume I*, pages 522–527, October 2012.
- [33] V. Bajpai and J. Schönwälder. A Survey on Internet Performance Measurement Platforms and Related Standardization Efforts. *IEEE Communications Surveys Tutorials*, 17(3), pages 1313-1341. September 2015.
- [34] C. G. Bampis, Z. Li, and A. C. Bovik. Continuous Prediction of Streaming Video QoE Using Dynamic Networks. *IEEE Signal Processing Letters*, 24(7) :1083–1087, July 2017.
- [35] S. Baraković and L. Skorin-Kapov. Survey and Challenges of QoE Management issues in Wireless Networks. *Journal of Computer Networks and Communications*, pages 1-28. Mars 2013.
- [36] L. Breiman. Bagging Predictors. *Machine Learning Journal*, Volume 24(2) : pages 123–140, August 1996.
- [37] L. Breiman. Random Forests. *Machine Learning Journal*, Volume 45(1) :pages 5–32, October 2001.

- [38] R. Bruzgiene, L. Narbutaite, T. Adomkus, and R. Cibulskis. Subjective and Objective MOS Evaluation of User's Perceived Quality Assessment for IPTV Service : a Study of the Experimental Investigations. In *Eletronika ir Eletrotechnika Journal, Volume 19, Number 7*, pages 110–113, June 2013.
- [39] M. Béri-Dexheimer, C. Bonnet, P. Chambon, K. Brochet, M.-J. Grégoire, and P. Jonveaux. L'hybridation Génomique Comparative sur Microréseau d'ADN (puces à l'ADN) en Pathologie Chromosomique Constitutionnelle. *Pathologie Biologie*, 55(1) :13 – 18, Février 2007.
- [40] C. Chen, X. Zhu, G. de Veciana, A. C. Bovik and R. W. Heath. Rate Adaptation and Admission Control for Video Transmission With Subjective Quality Constraints. *IEEE Journal of Selected Topics in Signal Processing*, 9(1) :22–36, February 2015.
- [41] C. Cousin, A. Victoria, S. Desmaris, S. Salaün, L. Petit and N. Bouy. . La Vidéo à la Demande par Abonnement en France : Marché et Stratégies des Acteurs. In *Conseil supérieur de l'audiovisuel, Direction des études, des affaires économiques, et de la prospective.*, pages 1-100. Janvier 2018.
- [42] C. Liu, I. Bouazizi and M. Gabbouj. Rate Adaptation for Adaptive HTTP Streaming. In *Proceedings of the Second Annual ACM Conference on Multimedia Systems (MMSys '11)*, pages 169-174. February 2011.
- [43] C. N. Pitas and D. E. Charilas and A. D. Panagopoulos and P. Chatzimisios and P. Constantinou. ANFIS-based Quality Prediction Models for AMR Telephony in Public 2G/3G Mobile Networks. In *2012 IEEE Global Communications Conference (GLOBECOM)*, pages 1728–1732, December 2012.
- [44] C. Zhou, C. W. Lin, X. Zhang and Z. Guo. Buffer-based Smooth Rate Adaptation for Dynamic HTTP Streaming. In *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, pages 1–9, October 2013.
- [45] P. Casas, A. D'Alconzo, F. Wamser, M. Seufert, B. Gardlo, A. Schwind, P. Tran-Gia, and R. Schatz. Predicting QoE in Cellular Networks using Machine Learning and in-

- Smartphone Measurements. In *2017 Ninth International Conference on Quality of Multimedia Experience (QoMEX)*, pages 1-6. May, 2017.
- [46] M. Champagne. Outils Non-paramètres de Détection de Ruptures Univariées et de Dépendance : Application en Hydrologie. In *Thèse de Doctorat Université du Québec à Trois-Rivières*, pages 1–137, Septembre 2009.
- [47] W. L. Chenwei Deng, L. Ma and L. N. Ngan. *Visual Signal Quality Assessment : Quality of Experience (QoE)*. Springer International Publishing, pages 1–320. November 2014.
- [48] W. Cherif. *Adaptation de contexte basée sur la Qualité d’Expérience dans les réseaux Internet du Futur*. Thèse Informatique, Université Rennes I, pages 1-138. Juin 2013.
- [49] H. Chouaib. Sélection de caractéristiques : Méthodes et applications. In *Thèse de Doctorat Université Paris Descartes*, pages 1–153, Juillet 2008.
- [50] Cisco Systems, Inc. Cisco Visual Networking Index : Forecast and Methodology, 2016–2021. pages 1–28, February 2017.
- [51] E. Danish and M. Alreshoodi. Quality Prediction for Video over Error-Prone Networks. *ACTION-TV Newsletter. Issue 4. pages 1-6.*, December 2015.
- [52] Dempster, A. P. Upper and Lower Probabilities induced by a Multivalued Mapping. *Annals of Mathematical Statistics Journal*, Volume 38(2), pages 325-339. February 1967.
- [53] M. Draxler and H. Karl. Cross-layer Scheduling for Multi-Quality Video Streaming in Cellular Wireless Networks. In *2013 9th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 1181–1186, July 2013.
- [54] E. Gary, F. Larry and R. Brandon. How to Choose Algorithms for Microsoft Azure Machine Learning. *Microsoft Azure Machine Learning Wiki*, April 2017, Access : <https://docs.microsoft.com/en-us/azure/machine-learning/machine-learning-algorithm-choice>.
- [55] E. Lefevre. Belief function combination and conflict management. *Information Fusion Journal*, Volume 3(2) : pages 149–162, June 2002.

- [56] A. Esteban, G. Matias, G. Noelia, and with contributions from Li Guo. Package Adabag : Applies Multiclass AdaBoost.M1, SAMME and Bagging. <https://cran.r-project.org/web/packages/adabag/>, October 2017.
- [57] ETSI TR 102 643. Human Factors (HF) : Quality of Experience (QoE) Requirements for Real-Time Communication Services. *Version : V1.0.1*, pages 1-37. December 2009.
- [58] G. B. Fabien Lauer. Méthodes SVM pour l'identification. *Journées Identification et Modélisation Expérimentale (JIME'2006)*, pages 1–6, November 2006.
- [59] M. C. Florea, J. Dezert, P. Valin, F. Smarandache, and A. Joussetme. Adaptive Combination Rule and Proportional Conflict Redistribution rule for Information Fusion. *Computing Research Repository (CoRR) Journal*, pages 1–8, April 2006.
- [60] Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In *Proceedings of the Thirteenth International Conference On Machine Learning*. Morgan Kaufmann Publisher, pages 148–156. July 1996.
- [61] J. H. Friedman. Stochastic Gradient Boosting. *Comput. Stat. Data Anal.*, Volume 38(4) :pages 367–378, February 2002.
- [62] H. Du, C. Guo, Y. Liu and Y. Liu. Research on relationship between QoE and QoS based on BP Neural Network. In *2009 IEEE International Conference on Network Infrastructure and Digital Content*, pages 312–315, November 2009.
- [63] H. J. Kim and D. H. Lee and J. M. Lee and K. H. Lee and W. Lyu and S. G. Choi. The QoE Evaluation Method through the QoS-QoE Correlation Model. In *2008 Fourth International Conference on Networked Computing and Advanced Information Management*, volume 2, pages 719–725, September 2008.
- [64] H. Liu and H. Motoda. *Feature Selection for Knowledge Discovery and Data Mining*. Kluwer Academic Publishers, pages 1-214. Mars 1998.
- [65] A. Hamam, M. Eid, A. El Saddik, and N. D. Georganas. *A Fuzzy Logic System for Evaluating Quality of Experience of Haptic-Based Applications*. Springer Berlin Heidelberg publisher, pages 129–138. June 2008.

- [66] B. Han, F. Qian, L. Ji, and V. Gopalakrishnan. Mp-dash : Adaptive video streaming over preference-aware multipath. In *Proceedings of the 12th International on Conference on Emerging Networking EXperiments and Technologies*, CoNEXT '16, pages 129–143, December 2016.
- [67] F. Harle. Détection de ruptures multiples dans des séries temporelles multivariées : Application a l'inférence de réseaux de dépendance. In *Thèse de Doctorat Université du Université Grenoble Alpes*, pages 1–233, Juin 2016.
- [68] V. Heidrich-meisner, M. Lauer, C. Igel, and M. Riedmiller. Reinforcement Learning in a Nutshell. In *15th European Symposium on Artificial Neural Networks (ESANN)*, pages 1-12. April 2007.
- [69] R. Hofstede, P. Čeleda, B. Trammell, I. Drago, R. Sadre, A. Sperotto, and A. Pras. Flow Monitoring Explained : From Packet Capture to Data Analysis With NetFlow and IPFIX. *IEEE Communications Surveys Tutorials*, 16(4), pages 2037-2064. September 2014.
- [70] Hossfeld, T., Hock, D., Tran-Gia, P., Tutschku, K., and Fiedler, M. Testing the IQX Hypothesis for Exponential Interdependency between QoS and QoE of Voice Codecs iLBC and G.711. In *18th ITC Specialist Seminar on Quality of Experience*, page 105–114, January 2008.
- [71] T. Hoßfeld, M. Seufert, M. Hirth, T. Zinner, P. Tran-Gia, and R. Schatz. Quantification of YouTube QoE via Crowdsourcing. In *2011 IEEE International Symposium on Multimedia*, pages 494-499. December 2011.
- [72] J. C. J. Benesty and Y. Huang. On the Importance of the Pearson Correlation Coefficient in Noise Reduction. *Audio, Speech, and Language Processing, IEEE Transactions on*, 16(4) : pages 757–765, May 2008.
- [73] Y. H. J. Benesty, J. Chen and I. Cohen. Pearson Correlation Coefficient. In *Noise Reduction in Speech Processing. Springer Topics in Signal Processing. Volume 2*, pages 1–4, Mars 2009.

- [74] J. De Vriendt and D. De Vleeschauwer and D. Robinson. Model for Estimating QoE of Video delivered using HTTP Adaptive Streaming. In *2013 IFIP/IEEE International Symposium on Integrated Network Management (IM 2013)*, pages 1288–1293, May 2013.
- [75] J. Jiang, V. Sekar and H. Zhang. Improving Fairness, Efficiency, and Stability in HTTP-based Adaptive Video Streaming with FESTIVE. In *Proceedings of the 8th International Conference on Emerging Networking Experiments and Technologies, CoNEXT '12*, pages 97–108, January 2014.
- [76] J. Korhonen, N. Burini, J. You and E. Nadernejad. How to Evaluate Objective Video Quality Metrics Reliably. In *2012 Fourth International Workshop on Quality of Multimedia Experience*, pages 57–62, July 2012.
- [77] J. Liu, C. Rosenberg, G. Simon and G. Texier. Optimal Delivery of Rate-Adaptive Streams in Underprovisioned Networks. *IEEE Journal on Selected Areas in Communications*, Volume 32(4) :706–718, April 2014.
- [78] J. Pokhrel, B. Wehbi, A. Morais, A. Cavalli and E. Allilaire. Estimation of QoE of Video Traffic using a Fuzzy Expert System. In *2013 IEEE 10th Consumer Communications and Networking Conference (CCNC)*, pages 224–229, January 2013.
- [79] J. Tang, S. Alelyani and H. Liu. Feature Selection for Classification : A Review. *Data Classification : Algorithms and Applications*. CRC Press, pages 1–37, July 2014.
- [80] J. S. R. Jang. ANFIS : Adaptive-Network-based Fuzzy Inference System. *IEEE Transactions on Systems, Man, and Cybernetics*, Volume 23(3) :665–685, May 1993.
- [81] V. Joseph and G. de Veciana. Nova : Qoe-driven optimization of dash-based video delivery in networks. In *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*, pages 82–90, April 2014.
- [82] P. Juluri, V. Tamarapalli, and D. Medhi. SARA : Segment Aware Rate Adaptation Algorithm for Dynamic Adaptive Streaming over HTTP. In *2015 IEEE International Conference on Communication Workshop (ICCW)*, pages 1765–1770, June 2015.

- [83] K. D. Singh, Y. Hadjadj-Aoul and G. Rubino. Quality of Experience Estimation for Adaptive HTTP/TCP Video Streaming Using H.264/AVC. In *2012 IEEE Consumer Communications and Networking Conference (CCNC)*, pages 127–131, January 2012.
- [84] K. De Moor, I. Ketyko, W. Joseph and T. Deryckere and L. De Marezn and L. Martens and G. Verleye. Proposed Framework for Evaluating Quality of Experience in a Mobile, Testbed-oriented Living Lab Setting. *Mob. Netw. Appl.*, Volume 15(3) :pages 378–391, June 2010.
- [85] K. Miller, E. Quacchio and G. Gennari and A. Wolisz. Adaptation Algorithm for Adaptive Streaming over HTTP. In *2012 19th International Packet Video Workshop (PV)*, pages 173–178, May 2012.
- [86] K. Sentz and S. Ferson. Combination of Evidence in Dempster-Shafer Theory. Technical report, pages 1-196. April 2002.
- [87] K. Tsiakas, M. Abujelala, A. Lioulemes and F. Makedon. An intelligent Interactive Learning and Adaptation Framework for Robot-based Vocational training. In *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–6, December 2016.
- [88] T. Kawano, K. Yamagishi, and T. Hayashi. Performance Comparison of Subjective Assessment Methods for 3D video Quality. In *2012 Fourth International Workshop on Quality of Multimedia Experience*, pages 218–223, July 2012.
- [89] Keimel, Christian. *Design of Video Quality Metrics with Multi-Way Data Analysis : A Data Driven Approach*. Springer Publishing Company, Incorporated, 1st edition, pages 1-240. December 2015.
- [90] A. Khan, L. Sun, and E. Ifeachor. Content-Based Video Quality Prediction for MPEG4 Video Streaming over Wireless Networks. *Journal of Multimedia*, 4(4) :228–239, August 2009.
- [91] G. Koch, U. Keschull, and W. Rosenstiel. Breakpoints and breakpoint detection in source level emulation. In *Proceedings of 9th International Symposium on Systems Synthesis*, pages 26–31, November 1996.

- [92] L. Amour and S. Souihi and S. Hoceini and A. Mellouk. An Open Source Platform for Perceived Video Quality Evaluation. In *Proceedings of the 11th ACM Symposium on QoS and Security for Wireless and Mobile Networks (Q2SWinet)*, pages 139–140, 2015.
- [93] L. Amour, S. Souihi and A. Mellouk. Access base de données QoE environnement contrôlé dataset : . <https://github.com/PhDAmour/QoEcontrlledLaboratory>, June, 2017.
- [94] L. Amour, S. Souihi and A. Mellouk. Plateforme de Réalisation de Test Subjectifs basée sur la Méthode ACR : . <https://github.com/PhDAmour/PythonQoEEvaluation>, May, 2015.
- [95] L. Amour, S. Souihi and A. Mellouk. Accès à la base de données projet PoQEMON : . <https://github.com/PhDAmour/QoEpoqemonDatabase>, May, 2016.
- [96] L. Amour, S. Souihi and A. Mellouk. Accès à la base de données QoE vidéo avec la norme Dash : . <https://github.com/PhDAmour/QoEdashjsDatabase>, November, 2016.
- [97] L. Amour, S. Souihi, M. S. Mushtaq, S. Hoceini and A. Mellouk. Perceived Video Quality Evaluation based on Interactive/Repulsive Relation between the QoE IFs. In *2016 IEEE International Conference on Communications(ICC)*, pages 1–7, May 2016.
- [98] L. Amour, S. Souihi, S. Hoceini and A. Mellouk. Building a Large Dataset for Model-based QoE Prediction in the Mobile Environment. In *Proceedings of the 18th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, pages 313–317, November 2015.
- [99] K. U. R. Laghari. On Quality of Experience (QoE) for Multimedia Services in Communication Ecosystem. In *Thèse de Doctorat conjoint Télécom Sud Paris - Université Pierre Marie Curie*, pages 1–209, October 2013.
- [100] A. Langley, A. Riddoch, A. Wilk, and etc. The QUIC Transport Protocol : Design and Internet-Scale Deployment. In *Proceedings of the Conference of the ACM Special Interest Group on Data Communication, SIGCOMM '17*, pages 183–196, August 2017.

- [101] J. Lassall. Étude de l’Influence de la Qualité Audiovisuelle sur la Qualité d’Expérience du Spectateur : Combinaison d’Indicateurs Subjectifs, Physiologiques et Oculaires . In *Thèse de Doctorat conjoint Télécom Bretagne et Université de Bretagne-Sud*, pages 1–343, October 2013.
- [102] P. Le Callet, S. Moller, and A. Perkis. Qualinet White Paper on Definitions of Quality of Experience (QoE) and Related Concepts. In *Output version of the Dagstuhl seminar 12181, European Network on Quality of Experience in Multimedia Systems and Services (COST Action IC 1003)*, pages 1-25. March 2013.
- [103] A. Liaw and M. Wiener. Classification and Regression by randomForest. <http://CRAN.R-project.org/doc/Rnews/>, 2002.
- [104] E. Liotou, D. Tsolkas, and N. Passas. A Roadmap on QoE Metrics and Models. In *2016 23rd International Conference on Telecommunications (ICT)*, pages 1–5, May 2016.
- [105] A. Liotta, V. Menkovski, G. Exarchakos, and A. C. Sánchez. Quality of experience models for multimedia streaming. *Int. J. Mob. Comput. Multimed. Commun.*, 2(4) :1–20, October 2010.
- [106] M. Alreshoodi and J. Woods. QoE Prediction Model Based on Fuzzy Logic System for Different Video Contents. In *2013 European Modelling Symposium*, pages 635–639, November 2013.
- [107] M. Alreshoodi and J. Woods. Survey on QoE/QoS Correlation Models For Multimedia Services. *International Journal of Distributed and Parallel Systems (IJDPS), CoRR*, Volume abs/1306.0221 :pages 378–391, June 2014.
- [108] M. Azumi, T. Kurosaka and M. Bandai. A QoE-Aware Quality-Level Switching Algorithm for Adaptive Video Streaming. In *2015 IEEE Global Communications Conference (GLOBECOM)*, pages 1–5, December 2015.
- [109] M. Claeys, S. Latré and J. Famaey and T. Wu and L. Van and F. De Turck. Design of a Q-learning-based Client Quality Selection Algorithm for HTTP Adaptive Video Streaming. In *Adaptive and Learning Agents Workshop*, pages 30–37, May 2013.

- [110] M. Fiedler, T. Hossfeld and P. Tran-Gia. A Generic Quantitative Relationship Between Quality of Experience and Quality of Service. *IEEE Network*, Volume 24(2) :36–41, March 2010.
- [111] M. Galar, A. Fernandez and E. Barrenechea and H. Bustince and F. Herrera. A Review on Ensembles for the Class Imbalance Problem : Bagging, Boosting, and Hybrid-Based Approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(4) :463–484, July 2012.
- [112] M. H. Pinson and S. Wolf. A New Standardized Method for Objectively Measuring Video Quality. In *IEEE Transactions on Broad-casting, Volume 50(3)*, pages 312–322, September 2004.
- [113] M. Hall and E. Frank, G. Holmes and B. Pfahringer and P. Reutemann and I. Witten. The WEKA Data Mining Software : An Update. *ACM SIGKDD explorations newsletter*, (1) :pages 10–18, June 2009.
- [114] Y. Mao, J. Zhang, and K. B. Letaief. A Lyapunov Optimization Approach for Green Cellular Networks With Hybrid Energy Supplies. *IEEE Journal on Selected Areas in Communications*, 33(12) :2463–2477, December 2015.
- [115] P. McLean. Apple launches HTTP Live Streaming standard in iPhone 3.0. *Apple Insider report. Retrieved 15 October 2011*, July 2009.
- [116] S. Moteau, F. Guillemin, and T. Houdoin. Correlation between QoS and QoE for HTTP YouTube Content in Orange Cellular Networks. In *2017 IEEE 9th Latin-American Conference on Communications (LATINCOM)*, pages 1–6, November 2017.
- [117] B. A. M.S. Mushtaq and A. Mellouk. Regulating QoE for Adaptive Video Streaming using BBF Method. In *2015 IEEE International Conference on Communications (ICC)*, pages 6855–6860, June 2015.
- [118] M.S. Mushtaq and B. Augustin and A. Mellouk. QoE : User Profile Analysis for Multimedia Services. In *IEEE International Conference on Communications (ICC)*, pages 2289–2294, June 2014.

- [119] Murphy, Catherine K. Combining Belief Functions when Evidence Conflicts. *Decision Support System Journal*, Volume 29(1) :pages 1–9, July 2000.
- [120] M. Mushtaq, B. Augustin, and A. Mellouk. HTTP Rate Adaptive Algorithm with High Bandwidth Utilization. In *10th International Conference on Network and Service Management (CNSM)*, pages 372–375, November 2014.
- [121] M. S. Mushtaq, B. Augustin, and A. Mellouk. Crowd-sourcing Framework to Assess QoE. In *2014 IEEE International Conference on Communications (ICC)*, pages 1705–1710, June 2014.
- [122] C. Müller, D. Renzi, S. Lederer, S. Battista, and C. Timmerer. Using Scalable Video Coding for Dynamic Adaptive Streaming over HTTP in Mobile environments. In *2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, pages 2208–2212, August 2012.
- [123] E. K. M. v. N.M Perrin, H. Xu and E. Tourajd. Multimodal Dataset for Assessment of Quality of Experience in Immersive Multimedia. In *Proceedings of the 23rd ACM International Conference on Multimedia, MM '15*, pages 1007-1010. October 2015.
- [124] Nokia. Quality of Experience (QoE) of Mobile Services : Can it be measured and improved? *White Paper from Nokia*, pages 1–8, October 2011. Access : http://www.nokia.com/nokia_com_1/operators/downloads/nokia_services/whitepaper_qoe_net.pdf.
- [125] D. A. Norman and S. W. Draper. *User Centered System Design; New Perspectives on Human-Computer Interaction*. L. Erlbaum Associates Inc. publisher, pages 1-512. January 1986.
- [126] M. Oche, R. M. Noor, and C. Chembe. Multivariate statistical approach for estimating qoe of real-time multimedia applications in vehicular its network. *Computer Communications Journal*, Volume 104 :pages 88–107, 2017.
- [127] P. Calyam, P. Chandrasekaran, G. Trueb, et al., Multi-Resolution Multimedia QoE Models for IPTV Applications. *International Journal of Digital Multimedia Broadcasting*, Volume 2012 :1–13, October 2012.

- [128] P. K. Mallapragada and R. Jin and A. K. Jain and Y. Liu. SemiBoost : Boosting for Semi-Supervised Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(11) :2000–2014, November 2009.
- [129] P. M. Arun Kumar and S. Chandramathi. Intelligent Video QoE Prediction Model for Errorprone Networks. In *Indian Journal of Science and Technology*, Volume 8(16), pages 1–9, July 2015.
- [130] P. Reichl and S. Egger and R. Schatz and A. D’Alconzo. The Logarithmic Nature of QoE and the Role of the Weber-Fechner Law in QoE Assessment. In *2010 IEEE International Conference on Communications*, pages 1–5, May 2010.
- [131] P. Smets. Quantifying Beliefs by Belief Functions : An Axiomatic Justification. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence.*, pages 598–605, August 1993.
- [132] P. Walley. *Statistical Reasoning with Imprecise Probabilities*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. Taylor & Francis Publisher, pages 1-700. December 1991.
- [133] A. Peters, T. Hothorn, B. D. Ripley, T. Therneau, and B. Atkinson. Improved Predictive Models by Indirect Classification and Bagging for Classification, Regression and Survival Problems as well as Resampling based Estimators of Prediction Error. <https://cran.r-project.org/web/packages/ipred/>, March 2017.
- [134] W. J. Priyadarshana and G. Sofronov. Multiple Break-Points Detection in Array CGH Data via the Cross-Entropy Method. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 12(2) :487–498, March 2015.
- [135] P. R. Bertrand, M. Fhima, and A. Guillin. Off-line Detection of Multiple Change Points with the Filtered Derivative with P-Value method. *Sequential Analysis*, pages 1–26, February 2011.
- [136] R. Stankiewicz, and A. Jajszczyk. A Survey of QoE Assurance in Converged Networks. *Comput. Netw.*, Volume 55(7) :pages 1459–1473, May 2011.

- [137] R. Streijl, C. Winkler, H. Stefan and S. David. Mean Opinion Score (MOS) Revisited : Methods and Applications, Limitations and Alternatives. *Multimedia System*, Volume 22(2) :pages 213–227, March 2016.
- [138] A. Raake, J. Gustafsson, S. Argyropoulos, M. N. Garcia, D. Lindgren, G. Heikkilä, M. Pettersson, P. List, and B. Feiten. IP-Based Mobile and Fixed Network Audiovisual Media Services. *IEEE Signal Processing Magazine*, 28(6), pages 68-79. November 2011.
- [139] Reuven Y. Rubinstein. Optimization of Computer Simulation Models with Rare Events. *European Journal of Operational Research*, Volume 99(1) :pages 89 – 112, May 1997.
- [140] F. Ribeiro, D. Florencio, and V. Nascimento. Crowdsourcing subjective image quality evaluation. In *2011 18th IEEE International Conference on Image Processing*, pages 3097–3100, September 2011.
- [141] G. Ridgeway. Package GBM : Generalized Boosted Regression Models. <https://cran.r-project.org/web/packages/gbm/>, March 2017.
- [142] H. Riiser, P. Vigmostad, C. Griwodz, and P. Halvorsen. Commute Path Bandwidth Traces from 3G Networks : Analysis and Applications. In *Proceedings of the 4th ACM Multimedia Systems Conference*, MMSys '13, pages 114–118, February 2013.
- [143] S. Aroussi and T. Bouabana-Tebibel and A. Mellouk. Empirical QoE/QoS Correlation Model based on Multiple Parameters for VoD flows. In *2012 IEEE Global Communications Conference (GLOBECOM)*, pages 1963–1968, December 2012.
- [144] S. Chen , C. Chu, S. Yeh, H. Chu and P. Huang . Modeling the QoE of Rate Changes in Skype/SILK VoIP Calls. *IEEE/ACM Trans. Netw.*, 22(6) :pages 1781–1793, December 2014.
- [145] S. García and J. Cabrera and N. García. Quality-Control Algorithm for Adaptive Streaming Services Over Wireless Channels. *IEEE Journal of Selected Topics in Signal Processing*, 9(1) :50–59, February 2015.
- [146] S. Glenn. *A Mathematical Theory of Evidence*. Princeton University Press, pages 1-314. April 1976.

- [147] S. Hemminger. Network Emulation with NetEm. In *Australia's 6th National Linux Conference (LCA 2005)*. Linux Australia pulisher, April 2005.
- [148] S. Ickin, K. Wac, M. Fiedler, L. Janowski, J. H. Hong and A. K. Dey. Factors Influencing Quality of Experience of Commonly used Mobile Applications. *IEEE Communications Magazine*, Volume 50(4) :48–56, April 2012.
- [149] S. Khirman and P. Henriksen. Relationship between Quality-of-Service and Quality-of-Experience for Public Internet Service. In *3rd Passive Active Measurement Workshop*, pages 1–6, Mars 2002.
- [150] S. Khorsandroo and R. Md Noor and S. Khorsandroo. A Generic Quantitative Relationship between Quality of Experience and Packet Loss in Video Streaming Services. In *2012 Fourth International Conference on Ubiquitous and Future Networks (ICUFN)*, pages 352–356, July 2012.
- [151] S. Khorsandroo, R. Noor, and S. Khorsandroo. The Role of Psychophysics Laws in Quality of Experience Assessment : A Video Streaming Case Study. In *Proceedings of the International Conference on Advances in Computing, Communications and Informatics, ICACCI '12*, pages 446–452, 2012.
- [152] S. Le Hegarat-Mascle and I. Bloch and D. Vidal-Madjar. Application of Dempster-Shafer Evidence Theory to Unsupervised Classification in Multisource Remote Sensing. *IEEE Transactions on Geoscience and Remote Sensing*, Volume 35(4) :1018–1031, July 1997.
- [153] R. E. Schapire. A Brief Introduction to Boosting. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI) - Volume 2*, pages 1401–1406, July 1999.
- [154] P. Smets. Belief Functions : The Disjunctive Rule of Combination and the Generalized Bayesian Theorem . *International Journal of Approximate Reasoning* , Volume 9(1) :pages 1–35, August 1993.
- [155] P. Smets. *The Transferable Belief Model for Quantified Belief Representation*. Springer Netherlands Publisher, pages 267–301. January 1998.

- [156] K. Spiteri, R. Urgaonkar, and R. K. Sitaraman. Bola : Near-optimal bitrate adaptation for online videos. In *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9, April 2016.
- [157] M. Suznjevic, J. Saldana, M. Matijasevic, and M. Vuga. Impact of Simplemux Traffic Optimisation on MMORPG QoE. *5th ISCA/DEGA Workshop on Perceptual Quality of Systems*, Publisher Zenodo. February 2017.
- [158] T. Chai and R. R. Draxler. Root Mean Square Error (RMSE) or Mean Absolute Error (MAE) ?. –Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*, Volume 7(3) :pages 1247–1250, May 2014.
- [159] T. Denoeux. A k-Nearest Neighbor Classification Rule based on Dempster-Shafer Theory. *IEEE Transactions on Systems, Man, and Cybernetics*, Volume 25(5) :804–813, May 1995.
- [160] T. Hoßfeld, M. Hirth, and J. Redi, F. Mazza, and ...eds. Best Practices and Recommendations for Crowdsourced QoE - Lessons learned from the Qualinet Task Force "Crowdsourcing". COST Action IC1003 European Network on Quality of Experience in Multimedia Systems and Services (QUALINET), pages 1-28. October 2014.
- [161] T. Hoßfeld, P.E Heegaard, M. Varela, and S. Möller. QoE beyond the MOS : An in-depth Look at QoE via Better Metrics and their Relation to MOS. *Quality and User Experience Journal*, Volume 1(1) :pages 1–2, September 2016.
- [162] T. Hoßfeld, S. Biedermann, R. Schatz, A. Platzer, S. Egger and M. Fiedler. The Memory Effect and its Implications on Web QoE Modeling. In *2011 23rd International Teletraffic Congress (ITC)*, pages 103–110, September 2011.
- [163] T. Nguyen and T. Vu, N.P. Ngoc and T.C. Thang. SDP-Based Quality Adaptation and Performance Prediction in Adaptive Streaming of VBR Videos *Advances in Multimedia. Hindawi Advances in Multimedia Journal*, Volume 7323681-2017 : pages 1–12, May 2017.

- [164] T. Rahrer (Nortel), R. Fiandra (FastWeb) and S. Wright (BellSouth). DSL Forum Technical Report TR-126 : Triple-Play Services, Quality of Experience (QoE) Requirements. pages 1–129, December 2006.
- [165] T. Spetebroot, S. Afra, N. Aguilera, D. Saucez and C. Barakat. From Network-level Measurements to expected Quality of Experience : the Skype Use Case. In *IEEE International Workshop on Measurement and Networking (M&N)*, pages 1-6. November 2015.
- [166] T. Wang, A. Pervez and H. Zou. VQM-based QoS/QoE Mapping for Streaming Video. In *2010 3rd IEEE International Conference on Broadband Network and Multimedia Technology (IC-BNMT)*, pages 807–812, October 2010.
- [167] T. Zhang, F. Ren, W., Cheng, X. Luo, R. Shu and X. Liu. Modeling and Analyzing the Influence of Chunk Size Variation on Bitrate Adaptation in DASH. In *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, pages 1–9, May 2017.
- [168] S. Tavakoli, K. Brunnström, J. Gutiérrez, and N. García. Quality of Experience of Adaptive Video Streaming : Investigation in Service Parameters and Subjective Quality Assessment Methodology. *Signal Processing : Image Communication Journal*, Volume 39 :pages 432–443, 2015.
- [169] J. S. Turner and D. E. Taylor. Diversifying the Internet. In *IEEE Global Telecommunications Conference (GLOBECOM)*, Volume 2, pages 760-766, December 2005.
- [170] T.Y Huang, R. Johari, N. McKeown, M. Trunnell and M. Watson. A Buffer-based Approach to Rate Adaptation : Evidence from a Large Video Streaming Service. In *Proceedings of the 2014 ACM Conference on SIGCOMM*, SIGCOMM '14, pages 187–198, August 2014.
- [171] V. A. Machado, C. N. Silva, R. S. Oliveira, A. M. Melo, M. Silva, C. R. L. Francês and J. C. W. A. Costa and N. L. Vijaykumar and C. M. Hirata. A New proposal to Provide Estimation of QoS and QoE over WiMAX Networks : An Approach based on Computational Intelligence and Discrete-Event Simulation. In *2011 IEEE Third Latin-American Conference on Communications*, pages 1–6, October 2011.

- [172] V. Krishnamoorthi, N. Carlsson, E. Halepovic and E. Petajan. BUFFEST : Predicting Buffer Conditions and Real-time Requirements of HTTP(S) Adaptive Streaming Clients. In *ACM SIGMM Conference on Multimedia Systems (MMSys)*, pages 1–12, June 2017.
- [173] V. Menkovski and A. Liotta. Intelligent Control for Adaptive Video Streaming. In *2013 IEEE International Conference on Consumer Electronics (ICCE)*, pages 127–128, January 2013.
- [174] V. Menkovski, G. Exarchakos, A. Liotta and A. C. Sánchez. Measuring Quality of Experience on a Commercial Mobile TV Platform. In *2010 Second International Conferences on Advances in Multimedia*, pages 33–38, June 2010.
- [175] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag New York Publisher, New York, NY, USA, pages 1-314. May 1995.
- [176] M. Vranješ, S. Rimac-Drlje, and K. Grgić. Review of Objective Video Quality Metrics and Performance Comparison Using Different Databases. *Image Commun.*, Volume 28(1) :pages 1–19, 2013.
- [177] X. Wang, A. Chowdhery, and M. Chiang. SkyEyes : Adaptive Video Streaming from UAVs. In *Proceedings of the 3rd Workshop on Hot Topics in Wireless (HotWireless)*, pages 2–6, October 2016.
- [178] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image Quality Assessment : From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing*, Volume 13(4) :600–612, April 2004.
- [179] S. Xiang, M. Xing, L. Cai, and J. Pan. Dynamic Rate Adaptation for Adaptive Video Streaming in Wireless Networks. *Image Commun. Journal*, Volume 39 :pages 305–315, November 2015.
- [180] Y. Ben Youssef, A. Mellouk, M. Afif and S. Tabbane. Video Quality Assessment Based on Statistical Selection Approach for QoE Factors Dependency. In *2016 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6, December 2016.

- [181] Y. Chen and F. Zhang and K. Wu and Q. Zhang. QoE-Aware Dynamic Video Rate Adaptation. In *2015 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6, December 2015.
- [182] Y. Chen, K. Wu and Q. Zhang. From QoS to QoE : A Tutorial on Video Quality Assessment. *IEEE Communications Surveys Tutorials*, Volume 17(2) :1126–1165, October 2015.
- [183] Y. Sani, A. Mauthe and C. Edwards and M. Mu. A bio-inspired HTTP-based Adaptive Streaming player. In *2016 IEEE International Conference on Multimedia Expo Workshops (ICMEW)*, pages 1–4, July 2016.
- [184] Y. Sun, X. Yin, J. Jiang, and V. Sekar, and F. Lin, and N. Wang, T. Liu and B. Sinopoli. CS2P : Improving Video Bitrate Selection and Adaptation with Data-Driven Throughput Prediction. In *Proceedings of the 2016 ACM SIGCOMM Conference (SIGCOMM)*, pages 272–285, August 2016.
- [185] H. Yang, X. Chen, X. Z. Z. Yang, and Y. Chen. Opportunities and Challenges of HTTP Adaptive Streaming. *International Journal of Future Generation Communication and Networking Volume 7(6)*, pages 165-180. June 2014.
- [186] X. Yin, A. Jindal, V. Sekar, and B. Sinopoli. A Control-Theoretic Approach for Dynamic Adaptive Video Streaming over HTTP. In *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*, pages 325–338, August 2015.
- [187] H. Ying. General Takagi-Sugeno Fuzzy Systems with Simplified Linear Rule Consequent are Universal Controllers, Models and Filters. *Information Sciences*, 108(1) :91 – 107, 1998.
- [188] Z-H. Zhou. *Ensemble Methods : Foundations and Algorithms*. Chapman & Hall/CRC Publisher, 1st edition, pages 1-236. June 2012.
- [189] A. H. Zahran, J. Quinlan, D. Raca, C. J. Sreenan, E. Halepovic, R. K. Sinha, R. Jana, and V. Gopalakrishnan. OSCAR : An Optimized Stall-cautious Adaptive Bitrate Streaming Algorithm for Mobile Networks. In *Proceedings of the 8th International Workshop on Mobile Video (MoVid)*, pages 1–6, May 2016.

- [190] H. Zhang and X. Jiang. A QoE-driven Approach to Rate Adaptation for Dynamic Adaptive Streaming over HTTP. In *2016 IEEE International Conference on Multimedia Expo Workshops (ICMEW)*, pages 1–6, July 2016.
- [191] H. Zimmermann. Innovations in Internetworking. chapter OSI Reference Model-The ISO Model of Architecture for Open Systems Interconnection, pages 2–9. Mars 1988.