

# Analyse de la qualité des liens Wi-Fi à partir de données expérimentales

Thierry Arrabal<sup>1</sup>, Marija Stojanova<sup>1</sup>, Isabelle Guérin Lassous<sup>1</sup> et Joris Picot<sup>1</sup>

<sup>1</sup>Univ Lyon, ENS de Lyon, Université Claude Bernard Lyon 1, Inria, CNRS, LIP

---

La technologie Wi-Fi est utilisée dans de nombreuses applications comme par exemple les flottes de drones. Afin d'assurer une bonne communication entre deux drones voisins, la plupart des solutions sur les flottes de drones cherchent à positionner les drones entre eux de telle sorte que la puissance reçue soit la meilleure possible. Or la technologie Wi-Fi a beaucoup évolué dernièrement et il n'est pas clair que l'utilisation de la seule puissance reçue soit suffisante pour déterminer la qualité de la communication Wi-Fi. Dans cet article, nous cherchons à mieux comprendre l'impact des paramètres Wi-Fi sur les performances des communications Wi-Fi. Notre étude se base sur une campagne de mesures via des expérimentations réelles et sur une étude d'analyse et d'apprentissage sur les données collectées<sup>†</sup>.

**Mots-clés :** qualité des liens, Wi-Fi, mesure, expérimentation, régression linéaire, apprentissage automatique

---

## 1 Introduction

La technologie Wi-Fi est de plus en plus envisagée dans les flottes de drones. Elle permet aux drones de communiquer et d'échanger des informations utiles au contrôle ainsi qu'à la mission de la flotte (p. ex. surveillance d'une zone géographique [6]). La majeure partie des travaux sur les flottes de drones reposent sur l'hypothèse que meilleure est la puissance du signal Wi-Fi reçu d'un drone voisin, meilleure sera la communication entre les deux drones (par exemple en termes de débit ou de taux de pertes) [4, 1, 5]. Certains travaux ont utilisé des modèles d'apprentissage automatique pour prédire la qualité du lien en fonction du RSSI, du SNR et de diverses autres informations sur l'état du lien [2, 5], mais ces travaux reposent sur des versions anciennes du Wi-Fi. Or cette dernière a beaucoup évolué ces dix dernières années avec, entre autres, de nombreuses évolutions au niveau de la couche physique (MCS, largeur de bande, MIMO, etc.) et l'utilisation d'algorithmes d'adaptation de débit souvent seulement connus des constructeurs de cartes Wi-Fi. Ces nombreuses métriques impactent les performances des communications Wi-Fi mais il n'existe pas d'étude qui identifie clairement et qualitativement cet impact avec les versions récentes du Wi-Fi.

Dans ce travail en cours, nous décrivons la méthodologie que nous avons utilisée pour mieux comprendre l'impact des métriques Wi-Fi sur les performances des communications Wi-Fi dans un contexte réaliste. Notre méthodologie repose sur deux étapes : i) une étape d'expérimentations permettant de collecter de nombreuses données sur les métriques et les performances Wi-Fi dans des environnements réels et avec des cartes Wi-Fi récentes (Section 2); ii) une étape d'analyse de données et d'apprentissage automatique permettant d'identifier les métriques clés ou inutiles, les corrélations entre les différentes métriques et permettant de dériver des modèles d'estimation des performances en fonction des métriques clés (Section 3). Dans cet article, nous nous sommes intéressés à la prédiction du taux de pertes. Nous concluons sur les premières leçons apprises grâce à cette méthodologie (Section 4).

## 2 Plateforme expérimentale et métriques collectées

### 2.1 Description de la plateforme expérimentale

La plateforme expérimentale est constituée de trois minis PC (nœuds) pilotés via un serveur de gestion. L'ensemble est coordonné via WalT, une plateforme logicielle permettant un déploiement simple d'expé-

---

<sup>†</sup>. Ce travail a reçu le soutien de l'ANR et de l'AID via le projet CONCERTO (ANR-20-ASTR-0003), ainsi que de la DGA et du CNRS.

riences réseaux<sup>‡</sup>. Les nœuds sont des *APU 4C4* équipés de carte Wi-Fi *Qualcomm Atheros QCA986x/988x 802.11ac* et disposent de 3 antennes. Le système d'exploitation utilisé est *Debian 10* (noyau *4.19.0-18-amd64*) et le pilote Wi-Fi *ath10k*. Le protocole 802.11ac (Wi-Fi 5) est utilisé dans la bande de fréquences des 5 GHz et configuré pour utiliser une largeur de bande maximale de 80 MHz.

Lors de la réception d'un paquet Wi-Fi, le système Debian ré-encapsule la trame 802.11 avec une en-tête Ethernet, ce qui empêche l'accès aux en-têtes physiques et MAC de la trame 802.11 lorsque qu'elle est prise en charge par le système. Il est toutefois possible de récupérer ces champs en configurant une carte Wi-Fi en mode monitor. Néanmoins, ce mode de fonctionnement ne peut pas être utilisé pour communiquer (envoyer ou recevoir des trames). Il est donc nécessaire d'utiliser un nœud dédié à cette fonction qui sera placé au plus proche du nœud pour lequel on souhaite récupérer des informations sur les trames qu'il reçoit afin de limiter les différences de condition de réception entre les deux.

## 2.2 Expérimentations réalisées

On s'intéresse ici à la qualité de la communication entre deux nœuds. Les expériences comprennent un nœud Source qui transmet du trafic au nœud Destination, au côté duquel est placé un nœud Monitor capturant (au mieux) les trames reçues par ce dernier.

Lors d'une expérience, le nœud Source envoie un trafic UDP à saturation avec des paquets de 3098 octets pendant 120 secondes au nœud Destination. Trois expériences sont réalisées pour chaque scénario. Un scénario est caractérisé par une distance entre la source et la destination et une position d'antennes, identique pour les deux nœuds, parmi quatre (cf. figure 1) : (A) *vertical* (perpendiculaires au sol); (B) *horizontal* (parallèles au sol); (C) *en éventail* et (D) *côté* (antennes toutes orientées vers le même côté).

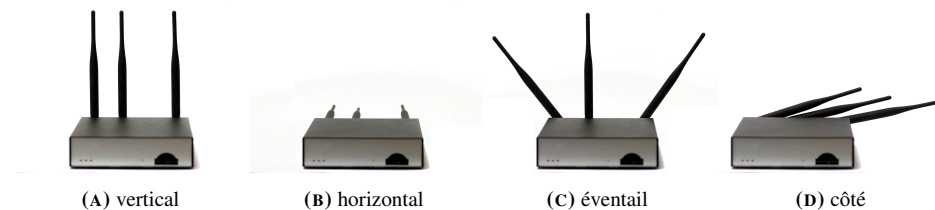


FIGURE 1 : Position d'antennes.

La distance séparant les deux nœuds varie de 0 mètre (les nœuds sont presque collés les uns aux autres) à 17 mètres. Les expériences se sont déroulées dans un couloir dans lequel les nœuds sont toujours en ligne de vue directe (pas d'obstacles). Le passage des personnes dans le couloir durant les expériences n'a pas pu être contrôlé, ce qui a permis d'apporter de la variabilité dans les conditions expérimentales.

Enfin, l'algorithme d'adaptation de débit utilisé par la carte Wi-Fi<sup>§</sup> n'a pas été désactivé. La carte et le pilote peuvent donc librement adapter les paramètres de communication en fonction des conditions radio. Les expériences se déroulent donc dans des conditions très proches d'un usage normal.

## 2.3 Métriques collectées et calculées

Pour notre étude, nous avons choisi de conserver les métriques suivantes : l'*indice de "Modulation and Coding Scheme" (MCS)*, le *nombre de flux spatiaux (NSS)*, la *largeur de bande utilisée (BW)*, la *puissance du signal reçu (RSSI)* et le *débit de transmission physique (datarate)*. À ces métriques, nous ajoutons les caractéristiques du scénario : *distance* et *orientation des antennes*.

Nous avons également ajouté le *nombre de trames reçues*, le *nombre de trames perdues*, le *taux de pertes* et le *débit utile*, calculés à partir de la trace collectée sur les trames reçues en utilisant le numéro de séquence des en-têtes MAC.

Ces métriques, appelées par la suite caractéristiques, sont moyennées sur des intervalles de 100 milli-secondes<sup>¶</sup>, constituant un jeu de données d'environ 180 000 observations qui est fourni aux algorithmes

‡. <https://walt-project.liglab.fr/>

§. sur la majeure partie des cartes, cet algorithme est inconnu des utilisateurs

¶. Un intervalle comprend entre 1000 et 2000 paquets.

d'apprentissage.

### 3 Apprentissage d'un modèle de prédiction du taux de pertes

Le processus d'apprentissage d'un modèle à partir de données peut être divisé en plusieurs parties : analyse des données, ajustement d'un ou plusieurs modèles aux données, prédiction des sorties pour de nouvelles données d'entrée [3]. Nous nous intéressons ici à la prédiction du taux de pertes défini comme le ratio entre le nombre de trames perdues et le nombre de trames envoyées.

#### 3.1 Analyse des données

Dans un premier temps, nous essayons de supprimer les caractéristiques qui peuvent l'être afin de réduire la complexité du modèle et d'éviter qu'il surajuste les données d'entraînement. Les régressions linéaire et polynomiale auxquelles nous nous intéressons ont toutes deux une solution de forme fermée utilisant l'équation normale dont la complexité est linéaire en nombre d'observations et polynomiale en nombre de caractéristiques. Après visualisation et analyse des distributions de toutes les caractéristiques, nous avons supprimé la largeur de bande (BW) et le nombre de flux spatiaux (NSS), car elles ont des valeurs respectives de 80 MHz et 3 flux dans environ 90% des cas. Elles ne peuvent donc pas être utilisées pour prédire le taux de perte ou toute autre caractéristique<sup>¶</sup>.

#### 3.2 Ajustement des modèles et prédiction du taux de perte

Nous avons principalement utilisé la bibliothèque scikit-learn [7]. Pour les modèles d'apprentissage, nous avons considéré les régressions Ridge dites offline \*\*, car leurs contraintes aident à traiter les problèmes de multicollinéarité potentiellement présents dans nos données et à éviter le surajustement aux données d'entraînement. Nous limitons le degré des modèles polynomiaux à 2, car un modèle quadratique semble bien s'ajuster aux données. Les ensembles d'entraînement et de test sont divisés avec un ratio de 80/20 avec une répartition stratifiée pour garantir que la distribution par distance des ensembles d'entraînement et de test est la même que celle de l'ensemble de données complet.

Ensemble des caractéristiques prises en compte	Modèle linéaire			Modèle polynomial		
	RMSE	MAE	R2	RMSE	MAE	R2
Toutes les caractéristiques	0,079	0,044	0,756	0,045	0,020	0,920
Nombre de trames, débit utile, MCS, RSSI	0,084	0,050	0,726	0,067	0,038	0,826

TABLE 1 : Erreurs et score R2 pour les modèles estimés.

Le premier modèle utilise toutes les caractéristiques disponibles à l'exception de BW et NSS. Les figures 2a et 2b montrent respectivement la précision de la prédiction pour les modèles Ridge linéaires et polynomiaux sur l'ensemble de test<sup>††</sup>. Les erreurs et le coefficient de détermination R2 correspondants sont résumés dans le tableau 1<sup>‡‡</sup>. Par la suite, nous nous concentrons uniquement sur le modèle polynomial, le modèle linéaire n'étant pas suffisamment précis.

Pour trouver l'ensemble minimal de caractéristiques permettant d'obtenir une prédiction suffisamment précise, nous avons cherché les caractéristiques plus importantes du modèle polynomial. Ces caractéristiques sont le nombre de trames reçues, le MCS, le RSSI et le débit utile. La figure 2c montre la prédiction de ce nouveau modèle se basant seulement sur la combinaison de ces caractéristiques et le tableau 1 résume ses erreurs et coefficient de détermination R2. Nous remarquons que si R2 est plus bas qu'avec toutes les caractéristiques prises en compte, les erreurs n'augmentent que légèrement. Cela signifie que le modèle se généralise toujours bien aux nouvelles données et est capable de faire des prédictions précises.

¶. Les figures ne sont pas présentées ici par souci de brièveté.

\*\*.. Les paramètres du modèle ne sont estimés qu'une seule fois avec toutes les données d'entraînement disponibles, et le modèle est ensuite utilisé sur les données de test.

††. Sur un ordinateur Dell Latitude 5420 (Processeur Intel® Core™ i5-1145G7, 16GB DDR4) le temps de prédiction pour une seule observation est inférieur à 1 micro seconde, quel que soit l'ensemble de caractéristiques.

‡‡. Erreur quadratique moyenne (RMSE, Root Mean Squared Error) et erreur absolue moyenne (MAS, Mean Absolute Error).

