

# Slotted Aloha à Première Transmission Différée : Une Nouvelle Solution pour Supporter les Applications Sensibles au Délai

Essaid Sabir<sup>1,2</sup>, Mohammed Raiss El-Fenni<sup>2</sup> et Mohamed Elkamili<sup>2,3</sup>

1 : LIA/CERI, Université d'Avignon, Agroparc B.P. 1228, Avignon, Cedex 9, France.

2 : LIMIARF, Université Mohammed V-Agdal, B.P. 1014 RP, 10000 Rabat, Maroc.

3 : LISQ, Faculté des Sciences Dhar El Mahraz, B.P. 1796 Atlas Fès, Maroc.

Contact :

essaid.sabir@univ-avignon.fr

raiss.med@gmail.com

melkamil@yahoo.fr

---

## Résumé

Nous considérons le lien montant d'un réseau cellulaire, tel que l'UMTS, où  $m$  mobiles transmettent à la station de base sur un canal commun via le protocole slotted aloha. Pour réduire le taux de collision, en particulier à moyen et fort trafic, nous proposons une nouvelle variante que nous appelons aloha différé et qui se base sur l'idée suivante : Contrairement au protocole standard où un nouveau paquet est immédiatement transmis, nous considérons que chaque nouveau paquet est transmis avec une probabilité  $p_a$ . Nous avons construit un modèle Markovien basé sur le processus du nombre de paquets en attente. Puis, nous avons calculé le débit ainsi que le délai moyens et les avons utilisés comme des objectifs à optimiser par les mobiles. Ensuite, nous avons étudié l'impact des différentes variables de décision sur le débit et le délai, ainsi que la stabilité de notre nouveau protocole. Des conclusions ont été tirées à partir des exemples numériques montrant ainsi l'efficacité de notre solution ainsi que son intérêt pour supporter les applications interactives et temps-réel sur cette famille de protocoles.

## Abstract

We consider the uplink case of a cellular system where  $m$  bufferless mobiles transmit over a common channel to a base station, using the slotted aloha medium access protocol. We study the performance of this system under a new transmission scheduling scheme. In contrast to slotted aloha, each new arrival packet is first transmitted at a probability  $p_a$  and retransmitted later, if needed, with probability  $q_r$ . Under this proposal, we analyze the cooperative team in which a common goal is jointly optimized. Later, we derive the average throughput and the expected delay and use them as the objectives to optimize. We further study the impact of parameters on the average throughput as well as the expected delay. Furthermore, we address the stability behavior of our new scheme. We finally carry out extensive numerical examples to illustrate the obtained gain which concerns, in particular, the expected delay minimization.

**Mots-clés :** Aloha discrétisé, Evaluation de performances, distribution stationnaire.

**Keywords:** Slotted aloha, Performance evaluation, Steady state.

---

## 1. Introduction

Aloha [1] et slotted aloha [11] sont sans aucun doute les protocoles les plus étudiés dans la littérature des systèmes de télécommunication. En effet, ils ont toujours été de grand intérêt et sont utilisés pour la compréhension et l'analyse des systèmes de transmission sans fil. Ces méthodes d'accès sont largement implémentées dans les réseaux satellitaires et la téléphonie mobile pour le transfert sporadique de paquets de données ou considérées comme des mécanismes de réservation de ressource. Dans aloha, les nouveaux paquets sont immédiatement transmis et si plusieurs

paquets sont envoyés simultanément par plus d'un mobile, ils entrent impérativement en collision. Après qu'un paquet est transmis, l'émetteur reçoit l'accusé de réception (ACK) sur l'état de la transmission (succès ou collision). Tous les paquets non reçus sont donc corrompus et seront retransmis après un certain temps aléatoire. Dans cet article, nous nous intéressons à slotted aloha, version améliorée inventée par Lawrence G. Roberts [11], dans lequel le temps est divisé en unités égales appelées slots. Le temps de vulnérabilité s'est donc vu réduit de moitié. La transmission a toujours lieu au début du slot, c-à-d les transmissions sont synchronisées avec l'horloge du système, et l'ACK sera reçu à la fin du slot courant. Les paquets entrés en collision sont mis en attente (nous les appelons au long de cet article les paquets backloggés) jusqu'à leurs retransmissions planifiées après un nombre aléatoire de slots.

Quand plusieurs mobiles partagent le même canal (fréquence, code...), un conflit surgit et les mobiles sont vus en compétition pour transmettre leurs paquets. Récemment, le comportement égoïste/individualiste (selfish behavior) des mobiles dans les protocoles MAC a été largement analysé à l'aide de la théorie des jeux avec tout son formalisme Mathématique et la puissance de ses concepts de solution. Il a été démontré dans [2, 3, 5] et [10] que l'individualisme des mobiles conduit souvent à une tragique inefficacité du réseau. Ceci conduit les mobiles à tomber dans une situation similaire au dilemme des prisonniers largement connu par les théoriciens des jeux et les économistes. Cela montre, en effet, que l'équilibre de Nash<sup>1</sup> n'est pas efficace dans certaines situations. Ainsi, une meilleure utilisation du système sans fil nécessiterait une coordination entre les mobiles en utilisant des messages de signalisation explicites ou le déploiement d'un mécanisme d'arbitrage. Ces solutions peuvent être irréaliste compte tenu de la nature distribuée et arbitraire de la topologie (due à la mobilité, fin/nouveaux appels, environnement, ...) des réseaux sans fil d'un côté, et de la surcharge éventuelle suite aux échanges de messages de signalisation. Pour parvenir à une bonne performance sans avoir recours aux mécanismes de coordination, les mobiles ont besoin de coopérer ou de respecter une certaine priorité/hierarchie [12]. Il est donc avantageux de mettre dans le système un ensemble de mobiles, dont la mission est de fournir des incitations pour les autres mobiles pour coopérer et/ou respecter la priorité définie. Ce dernier mécanisme peut limiter le niveau d'agressivité (accès au canal) des mobiles et de résoudre le problème de compétition. Une autre façon de réduire l'accès concurrentiel serait de définir un coût de transmission (tarification de transmission). En effet, il a été démontré dans [2] que les coûts ont un effet stabilisateur, les mobiles auraient tendance à différer la transmission de paquets quand le coût de la transmission est élevé. Hélas, agir ainsi risque de faire augmenter le délai d'attente et la probabilité de blocage. Les auteurs de [12], ont montré que l'introduction de la hiérarchie entre mobiles améliore les performances de l'ensemble du système, et ce en fonction du trafic offert. À faible trafic, cette amélioration est due au fait que les mobiles retransmettent avec une plus grande probabilité comparé au slotted aloha, d'où le temps de backoff réduit. À moyenne et à forte charges, les mobiles ou plus précisément les *leaders* (maîtres) sont généralement moins agressifs que le cas de l'équilibre de Nash. De cette façon, la probabilité de collision est réduite d'où l'impact positif de la hiérarchie sur les performances du groupe des *leaders* et celui des *followers* (esclaves). Un phénomène intéressant est que lorsque le nombre de mobiles de type *followers* est plus grand que le nombre des *leaders*, ces derniers deviennent plus pacifiques et réduisent leur probabilité de retransmission, tandis que les autres deviennent trop agressifs et transmettent à une probabilité proche de 1. Plusieurs auteurs ont présenté des algorithmes d'accès au canal où plusieurs niveaux de puissance peuvent être utilisés. Ainsi un mobile réussira sa transmission quand il transmet à la plus grande puissance comparée à celles choisies par les autres mobiles se trouvant dans la cellule [2]. Par contre, dans [5], un mobile réussit sa transmission s'il choisit un niveau de puissance plus grand que ceux choisis par les autres mobiles tout en ayant un rapport signal sur bruit supérieur à un seuil donnée. Nous développons dans cet article un protocole à première transmission différée similaire au CSMA (Carrier Sense Multiple Access), il a l'avantage de ne pas avoir besoin d'écouter le canal ni de suivre l'algorithme de backoff pour les futures transmissions. Ceci permettrait donc de ne pas consommer de bande passante supplémentaire.

La suite de l'article est organisée comme suit : La Section 2 présente la formulation du problème et

<sup>1</sup> Un équilibre de Nash est un profil où aucun joueur ne peut améliorer son gain par une déviation unilatérale.

développe le modèle Markovien associé. Nous calculons les métriques de performance, étudions l'existence d'une solution ainsi que la stabilité de cette nouvelle variante de slotted aloha dans de la Section 3. Ensuite, nous discutons quelques exemples numériques dans la Section 4. L'article se termine par une conclusion et quelques perspectives.

## 2. Formulation du problème

Nous considérons un canal à collision partagé par  $m$  mobiles sans *buffer* et transmettant à un récepteur central, c-à-d, les mobiles ne génèrent pas de nouveaux paquets jusqu'à ce que l'actuel est transmis avec succès. Cette hypothèse est bien entendu réaliste dans le contexte de la signalisation ; En effet, il est naturel de supposer que la source ne doit pas commencer à générer de nouveaux paquets de signalisation (par exemple, une nouvelle réservation) tant que le paquet en cours n'est pas transmis avec succès. Par la suite, nous étudions slotted aloha comme un problème d'optimisation (problème d'équipe), où tous les mobiles cherchent à optimiser la même fonction objective (maximiser le débit totale ou minimiser le délai moyen). Nous définissons le mécanisme de priorité suivant : Chaque mobile  $i$  transmet ses nouveaux paquets avec la probabilité  $p_a^i$  alors qu'il retransmet ses paquets backloggés avec la probabilité  $q_r^i$  (il s'agit bien d'un processus de Bernoulli de paramètre  $q_r^i$ ) ; Ce nouveau mécanisme permet donc de 1) mieux gérer l'ordonnement des transmissions, 2) définir la priorité des paquets selon l'application sollicitée et le trafic du système et 3) soulager la charge instantanée du système. Pour étudier cette nouvelle méthode d'accès au canal, nous étendons le modèle markovien introduit pour la première fois dans [4]. La probabilité d'arrivée des paquets à la source  $i$  suit un processus de Bernoulli de paramètre  $q_a$  (c-à-d, au début de chaque slot, un nouveau paquet arrive à une source libre avec probabilité  $q_a$ ). Tant qu'une source ne réussit pas la transmission du paquet en cours, les nouveaux arrivés seront bloqués et perdus (parce que nous considérons des sources sans *buffer*). Il en résulte que les arrivés sont indépendants.

Nous considérons comme état globale du système, le processus stochastique qui représente le nombre de paquets backloggés noté par  $n$ . Soit  $\bar{q}_r$  (respectivement  $\bar{p}_a$ ) le vecteur des probabilités de retransmission (respectivement première transmission) de tous les mobiles, la  $j^{\text{ème}}$  composante est  $q_r^j$  (respectivement  $p_a^i$ ). Soit  $\pi(\bar{q}_r, \bar{p}_a)$  la distribution stationnaire de la chaîne de Markov où la  $n^{\text{ème}}$  composante  $\pi_n(\bar{q}_r, \bar{p}_a)$  représente la probabilité que  $n$  paquets soient backloggés. La solution de notre problème consiste donc à trouver le couple de vecteurs  $(\bar{q}_r^*, \bar{p}_a^*)$  qui maximise la fonction objective. Il s'agit clairement d'un problème d'optimisation vectorielle. Sans perdre de généralité et uniquement pour le but de garder la faisabilité du calcul, nous nous limitons dans notre problème au cas symétrique où tous les mobiles utilisent la même politique de transmission  $(q_r^i, p_a^i) := (q_r, p_a)$ . Nous devons donc trouver une solution symétrique optimale, c-à-d, un couple de probabilités de (re)transmission  $(q_r, p_a)$  qui ne dépend pas de  $i$ . Quand toutes les composantes de  $\bar{q}_r$  sont les mêmes, dit  $q$ , nous allons écrire (avec quelques abus de notation)  $\pi(q_r, p_a)$  au lieu de  $\pi(\bar{q}_r, \bar{p}_a)$ .

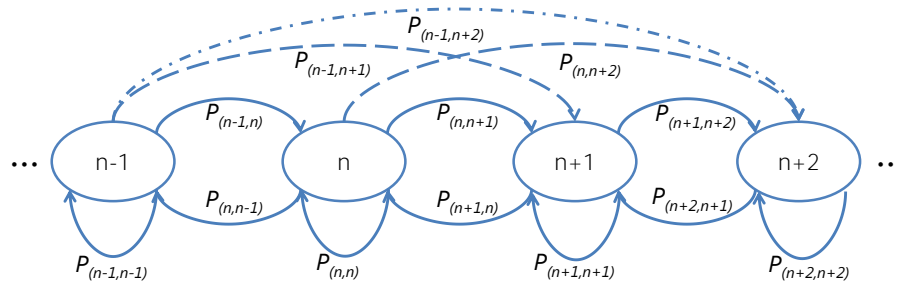


FIG. 1 – Chaîne de Markov globale du système. L'état du système est le nombre de paquets backloggés (backlog state), il peut diminuer d'au plus un (transition vers la gauche), mais peut augmenter par un nombre inférieur ou égal  $m - n$  (transition vers la droite).

**Flux d'arrivée et de retransmission :** Soit  $Q_r(i, n)$  la probabilité que  $i$  mobiles parmi  $n$  paquets en attente retransmettent dans le slot courant. Soit  $Q_a(i, n)$  la probabilité que  $i$  sources libres (avec

$i \leq m - n$ ) gènèrent des paquets dans le slot courant (c-à-d la probabilité que  $i$  nouveaux arrivés se sont produites à des sources qui n'ont pas de paquet en attente). De même, soit  $A_p(j, i)$  la probabilité que  $t$  nouveaux arrivés sont transmis directement dans le début du slot courant. Nous avons

$$Q_r(i, n) = \binom{n}{i} (1 - q_r)^{n-i} (q_r)^i. \quad (1)$$

$$Q_a(i, n) = \binom{m-n}{i} (1 - q_a)^{m-n-i} (q_a)^i. \quad (2)$$

$$A_p(j, i) = \binom{i}{j} (1 - p_a)^{i-j} (p_a)^j. \quad (3)$$

Posons  $Q_r(i, 0) = 0$ ,  $Q_a(i, m) = 0$  and  $A_p(j, 0) = 0$ .

La probabilité que le système transite d'un état  $n$  vers l'état  $n + i$  est donnée par

$$P_{n, n+i} = \begin{cases} Q_a(m-n, n) \sum_{\substack{j=0, t=0 \\ j+t \geq 2}}^{n, i} Q_r(j, n) A_p(t, i), & i = m - n, i \geq 2 \\ Q_a(i, n) \sum_{\substack{j=0, t=0 \\ j+t \geq 2}}^{n, i} Q_r(j, n) A_p(t, i) + \\ Q_a(i+1, n) \left[ Q_r(1, n) A_p(0, i+1) + Q_r(0, n) A_p(1, i+1) \right], & 0 \leq i < m - n \\ Q_a(0, n) Q_r(1, n), & i = -1 \end{cases} \quad (4)$$

### 3. Débit maximum et délai minimum

#### 3.1. Les métriques de performance

Nous passons maintenant à expliciter quelques indicateurs de performance (débit moyen et délai d'attente) comme fonctions de l'état stationnaire de la chaîne de Markov. Notons par  $\pi_n(q_r, p_a)$  la probabilité à l'équilibre que le système soit dans l'état  $n$  (état d'attente au début d'un slot). La solution optimale est obtenue par la résolution du problème d'optimisation suivant

$$\max_{q_r, p_a} \text{objective}(q_r, p_a) \text{ s.t. } \begin{cases} \pi(q_r, p_a) = \pi(q_r, p_a) \cdot P(q_r, p_a), \\ \sum_{n=0}^m \pi_n(q_r, p_a) = 1, \\ \pi_n(q_r, p_a) \geq 0, \quad n = 0, 1, \dots, m. \end{cases} \quad (5)$$

où  $\text{objective}(q_r, p_a)$  est remplacé par le débit moyen ou l'opposé du délai d'attente moyen. Nous constatons que la solution peut être obtenue en calculant récursivement l'état stationnaire des probabilités, comme dans le Problème 4.1 dans [4].

**Singularité en  $q_r = 0$  :** Le seul point où la chaîne de Markov  $P$  n'a pas de distribution stationnaire unique est en  $q_r = 0$ , où elle a un état absorbant :  $n = m$  (nous rappelons qu'en  $q_r = 0$ , slotted aloha a deux états absorbants  $n = m$  et  $n = m - 1$  [2, 3, 5]). Tous les autres Etats sont transitoires (pour toute  $q_a > 0$ ), et la probabilité de terminer à l'un des états absorbants dépend de la distribution initiale de la chaîne de Markov. Quand l'état  $m$  est atteint, le débit est égal à 0, ce qui signifie qu'il s'agit d'un état d'impasse (deadlock). Pour tout  $q_a > 0$  et  $q_r = 0$ , cet état est atteint avec une probabilité positive à partir de n'importe quel autre état initial, c'est pour cette raison que nous allons exclure le cas de  $q_r = 0$  et optimiser uniquement sur l'intervalle  $\epsilon < q_r \leq 1$ .

**Existence d'une solution :** La distribution stationnaire  $\pi(q_r, p_a)$  est continue sur  $0 < q_r \leq 1$  qui n'est pas un intervalle fermé, donc une solution peut ne pas exister. Comme nous nous limitons à l'intervalle fermé  $[\epsilon, 1]$ , où  $\epsilon > 0$ , une solution optimale existe toujours. Par conséquent, pour tout  $\gamma > 0$ , il existe une certaine  $q_r^* > 0$ , qui est  $\gamma$ -optimale.  $q_r^* > 0$  est dite  $\gamma$ -optimale pour la maximisation du débit, si elle satisfait  $T(q_r^*) \geq \text{thp}(q_r, p_a) - \gamma$  pour toute valeur de  $q_r \in [\epsilon, 1]$ . La même définition reste valable pour toute autre fonction objective (en particulier la minimisation du délai moyen). La résolution du système (5) fournit la distribution stationnaire qui nous

permettra d'en dériver les autres métriques de mesures.

Le nombre moyen de paquets backloggés est simplement calculé par

$$S(q_r, p_a) = \sum_{n=0}^m \pi_n(q_r, p_a) \cdot n. \quad (6)$$

De même, le débit total moyen (défini comme étant le nombre moyen de paquets qui sont transmis avec succès) est presque certainement donné par la constante

$$\begin{cases} \sum_{n=0}^m \sum_{i=0}^{m-n} \sum_{j=0}^{\min(1,n)} \sum_{t=0}^{\min(1,i)} \pi_n(q_r, p_a) Q_a(i, n) Q_r(j, n) A_p(t, i+1) \cdot \delta(1-t-j), & \text{Aloha différé} \\ \sum_{n=0}^m \pi_n(q_r, p_a) \left[ Q_a(0, n) Q_r(1, n) + Q_a(1, n) Q_r(0, n) \right]. & \text{Slotted aloha} \end{cases}$$

$\delta(\cdot)$  étant la distribution de Dirac. En utilisant l'équation de balance (c-à-d, l'entrée = la sortie) à l'équilibre, le débit total moyen satisfait (et donc peut être calculé plus facilement par)

$$T(q_r, p_a) = q_a \sum_{n=0}^m \pi_n(q_r, p_a) (m-n) = q_a (m - S(q_r, p_a)). \quad (7)$$

Le nombre moyen de paquets présents dans le système est égal à  $S(q_r, p_a) + T(q_r, p_a)$  (somme des paquets backloggés et des nouveaux arrivés). Quant au taux de départ, il est exactement égale au débit moyen  $\text{thp}(q_r, p_a)$ . Nous définissons le délai moyen de transmission d'un paquet  $D$ , comme étant le temps moyen, en slots, que prend un paquet de sa source à sa destination. L'application de la formule de Little donne

$$\begin{aligned} D(q_r, p_a) &= \frac{T(q_r, p_a) + S(q_r, p_a)}{T(q_r, p_a)}, \\ &= 1 + \frac{S(q_r, p_a)}{T(q_r, p_a)}, \\ &= 1 + \frac{S(q_r, p_a)}{q_a (m - S(q_r, p_a))}. \end{aligned} \quad (8)$$

En analysant les équations (7) et (8), nous constatons que maximiser le débit est équivalent à minimiser le délai moyen de transmission de paquets. Ainsi nous allons nous restreindre à maximiser le débit total moyen. Cependant, nous allons considérer le délai des paquets backloggés comme une autre fonctions objective à optimiser.

### 3.2. Mesure de performance des paquets backloggés

L'analyse du système basée sur la capacité de servir les paquets en attente de retransmission représente en effet une alternative intéressante pour la mesure des performances du système. Elle mesure la capacité du système à servir les paquets déjà entrés dans le système, d'où son intérêt surtout pour les applications de type temps-réel. Soit  $\Delta(q_r, p_a)$  le débit moyen des nouveaux arrivés, c-à-d, le nombre moyen des arrivés dont la première tentative est couronnée de succès. Alors, le débit moyen des paquets backloggés est donnée par :  $\tilde{T}(q_r, p_a) = T(q_r, p_a) - \Delta(q_r, p_a)$  où  $\Delta(q_r, p_a)$  est calculé à partir de la chaîne de Markov.

$$\Delta(q_r, p_a) = \begin{cases} \sum_{n=0}^m \sum_{i=1}^{m-n} \pi_n(q_r, p_a) Q_a(i, n) Q_r(0, n) A_p(1, i), & \text{Aloha différé} \\ \sum_{n=0}^m \pi_n(q_r, p_a) Q_a(1, n) Q_r(0, n). & \text{Slotted aloha} \end{cases} \quad (9)$$

Le délai des paquets backloggés  $\tilde{D}$ , qui est défini comme étant le temps moyen en slots, qu'un paquet en attente prend pour aller de la source à la destination, peut également être calculé à l'aide la formule de Little.

$$\tilde{D}(q_r, p_a) = 1 + \frac{S(q_r, p_a)}{\tilde{T}(q_r, p_a)}. \quad (10)$$

### 3.3. Stabilité

Une approche qualitative d'aborder les performances de notre nouveau protocole consiste à étudier sa stabilité. Slotted aloha est connu par son comportement bi-stable, nous nous posons donc les questions suivantes : Est ce que notre variante de slotted aloha souffre du même problème de bi-stabilité ? Si oui, y-a-t'il une différence entre les deux méthodes d'accès ? Si non, sous quelles conditions notre protocole serait meilleur que slotted aloha ?

Définissons  $p_n^{\text{succ}}$  comme étant la probabilité de réussir une transmission dans un slot quand le système est dans l'état  $n$ . Sur la base de la chaînes de Markov,  $p_n^{\text{succ}}$  est donnée par

$$p_n^{\text{succ}}(q_r, p_a) = \begin{cases} \sum_{i=0}^{m-n} \sum_{j=0}^{\min(1,n)} \sum_{t=0}^{\min(1,i)} Q_a(i, n) Q_r(j, n) A_p(t, i) \cdot \delta(1 - t - j), & \text{Aloha différé} \\ Q_a(0, n) Q_r(1, n) + Q_a(1, n) Q_r(0, n). & \text{Slotted aloha} \end{cases}$$

Définissons maintenant  $D_n$  dans l'état  $n$ , comme étant le changement de backlog d'un slot au suivant (le drift), qui est le nombre des arrivés, c-à-d,  $q_a(m - n)$ , moins les nombre de départs avec succès  $p_n^{\text{succ}}$ , qui est

$$D_n = q_a(m - n) - p_n^{\text{succ}}. \quad (11)$$

Dans slotted aloha standard, il a été démontré que trois équilibres existent. Le premier équilibre est efficace car il correspond à un nombre de paquets backloggés minime. le deuxième est instable et permet au système de passer d'un équilibre à l'autre. Par contre le troisième équilibre est caractérisé par un nombre de paquets en attente  $n$  très important (potentiellement  $n = m$ ), et donc correspond à une situation de congestion (faible débit et grand délai) ce qui réduit l'efficacité du système. Les points d'équilibre du système se produisent quand le drift s'annule, c-à-d la courbe  $p_n^{\text{succ}}$  et la droite  $q_a(m - n)$  se croisent. Quand le drift est positif l'état du système tend à augmenter, parce que le taux d'entrée au système est supérieur à son taux de sortie, et décroît lorsque la drift est négatif. Cela explique pourquoi le point d'équilibre du milieu est instable et les deux autres sont stables. Une situation bi-stable comme dans aloha standard n'est donc pas souhaitable car elle signifie, en pratique, que le système risque d'être absorbé par l'équilibre stable inefficace.

## 4. Résultats numériques

Nous présentons ici quelques exemples numériques évaluant les performances de la nouvelle solution d'accès au canal en prenant slotted aloha comme référence. Nous résolvons le système (5) pour  $m = 3$  et  $m = 10$  mobiles. Le délai moyen des paquets backloggés étant la fonction objective,

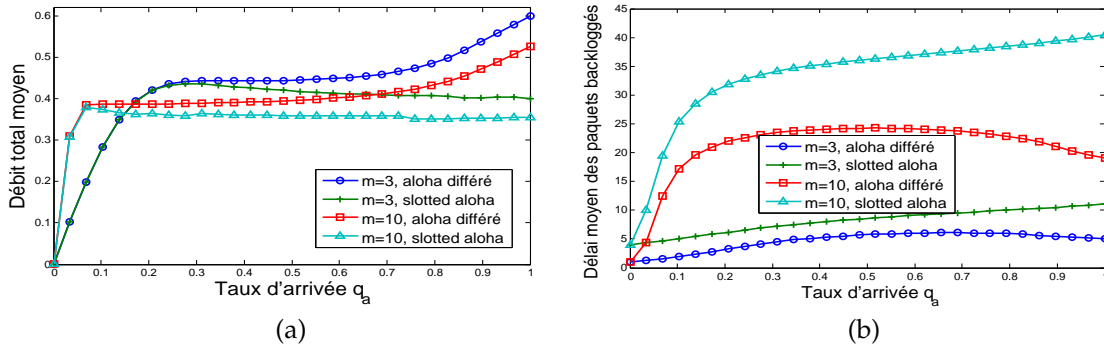


FIG. 2 – Débit total (a) et délai moyen des paquets backloggés quand le délai est minimisé (b).

nous constatons que le débit, Fig. 2 (a), et le délai moyen, Fig. 2 (b) se sont nettement améliorés en particulier à moyen et fort trafic. Ce résultat peut être expliqué en analysant la politique de transmission optimale ( $q_r, p_a$ ). En effet, à faible trafic les mobiles retransmettent avec une grande probabilité  $q_r$  tout en favorisant la transmission des nouveaux paquets. A fort trafic, la station de base recommande aux mobiles de diminuer leurs probabilités de retransmission ( $q_r, p_a$ ), Fig. 3, ce qui réduit le nombre de collisions et par la suite le débit continue à augmenter quand le trafic

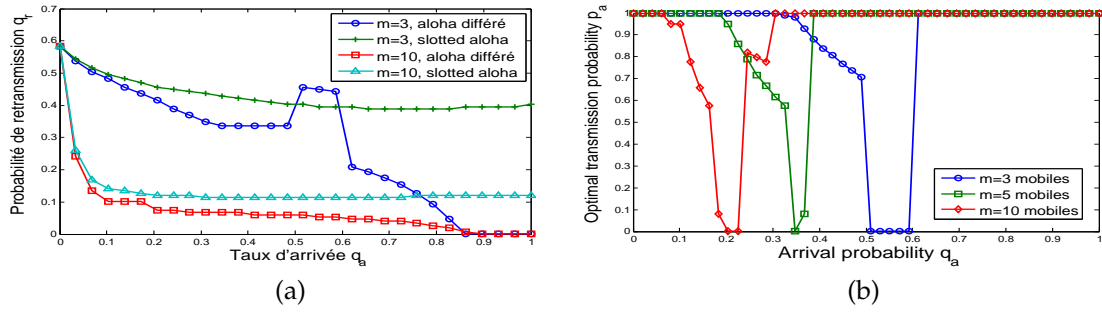


FIG. 3 – Probabilités de retransmission optimales quand le délai est minimisé (a). Un résultat similaire est obtenu le débit total est pris comme fonctions objective (b), notons que les mobiles sont légèrement moins agressifs.

devient élevé. Quand le débit total est considéré comme fonction à optimiser, nous remarquons que le débit de notre modèle coïncide avec celui de slotted aloha. Par contre, le délai d'attente moyen d'aloha différé s'avère être de loin inférieur à celui garanti par slotted aloha. En effet, le délai moyen correspondant à aloha différé reste quasiment constant indépendamment du trafic du système  $q_a$ , par contre le délai de slotted aloha a tendance à augmenter de façon exponentielle quand la trafic devient moyen ou élevée. Ceci (comportement d'aloha différé) est dû principalement au fait que lorsque  $q_a$  augmente,  $q_r$  diminue (c à d qu'on donne plus de priorité aux nouveaux paquets, d'où un nombre petit de baquets backloggés. Ce qui nous mène à dire que notre modèle est plus adapté aux services et applications temps-réel qui exigent des contraintes sur le délai d'attente.

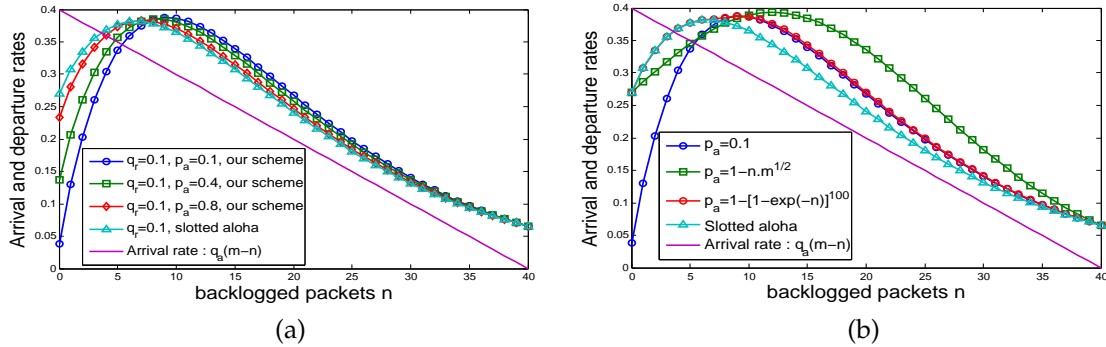


FIG. 4 – Stabilité de notre modèle et celle de slotted aloha pour les  $p_a$  statique (a) et  $p_a(n)$  dynamique (b).

Nous passons maintenant à l'illustration de la stabilité d'aloha différé. La figure 4 (a) montre le comportement de notre modèle par rapport à celui de slotted aloha standard pour  $q_a = 0.01$  et  $q_r = 0.1$ . Pour  $p_a$  statique, nous notons qu'à faible charge (taux de saturation), aloha est meilleur qu'aloha différé. En effet, quand le nombre moyen de paquets backloggés est petit, c-à-d seuls quelques mobiles ont des paquets à transmettre, le fait de différer les nouvelles transmissions réduit le taux de succès. Par contre quand le nombre de source saturés devient grand, aloha différé devient plus performant puisque le mécanisme de différer les nouvelle transmissions permet d'éviter/réduire les collisions. La figure 4 (b) montre le même schéma avec  $p_a$  dynamique. Nous remarquons que les performances sont améliorées quand  $p_a \rightarrow 1$  à faible charge et quand  $p_a \rightarrow 0$  à moyenne et forte charge. Cette constatation nous a inspiré de proposer une sélection dynamique du paramètre  $p_a$  telles les fonctions sigmoïdales (en forme de 'S') de type  $1 - (1 - e^{-n})^M$  avec  $M$  étant une constante donnée contrôlant la vitesse de la fonction  $p_a(n)$ . Le problème majeur avec ce mécanisme adaptatif est que les mobiles doivent estimer la valeur instantanée du nombre de paquets backloggés  $n$ . Ceci peut être facilement résolu à l'aide d'un algorithme d'apprentissage en se basant sur l'observation des événements "succès", "collision" et "libre", Problème 4.1 [4], donc

$$\hat{n}_{k+1} = \begin{cases} \max(\lambda, \hat{n}_k + \lambda - 1), & \text{libre ou succès} \\ \hat{n}_k + \lambda + (e - 2)^{-1}, & \text{collision.} \end{cases} \quad \text{avec } \lambda = q_a/m. \quad (12)$$

Ainsi le cas dynamique d'aloha différé reste presque toujours au dessus de slotted aloha pour

tout état  $n$  du système. A faible  $q_r$ , il est clair qu'aucune des deux variantes de slotted aloha ne souffrent du problème de bi-stabilité. Donc les deux schémas ne peuvent qu'être absorbés par le point d'équilibre efficace, c-à-d l'équilibre non congestionné.

La figure ci-contre montre l'impact de la probabilité de retransmission  $q_r$  sur la stabilité du protocole (ici  $q_r=0.5$ ). Globalement, nous affirmons que notre protocole est aussi bi-stable sauf qu'il est plus performant en terme de probabilité de succès. Un moyen de remédier au problème de bi-stabilité serait de définir un contrôle dynamique sur  $q_r$  en fonction du nombre instantané de paquets backloggés. Introduire un coût de transmission (consommation en énergie) est aussi une solution élégante pour que les mobiles réduisent leurs taux de transmission pour mieux gérer leur budget énergétique.

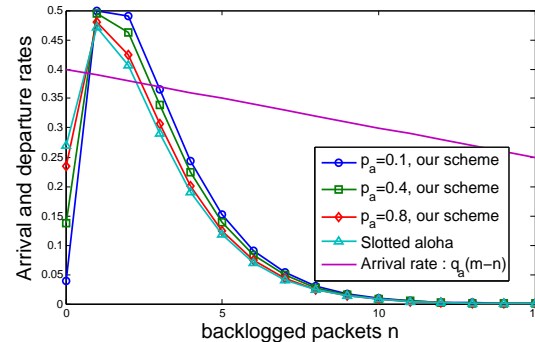


FIG. 5 – Cas où tous les schémas sont instables.

## 5. Conclusion

Nous avons proposé une version améliorée du protocole slotted aloha basée sur la transmission des nouveaux paquets avec une probabilité  $p_a$  au lieu d'une transmission immédiate. Ce mécanisme permet en particulier de mieux gérer l'ordonnement des transmissions en réduisant le nombre de collisions dans le système. Nous avons construit un modèle markovien et avons calculé la distribution stationnaire du processus associé. Cela nous a permis de calculer les différentes métriques de performance telles que le débit total, le délai moyen et la probabilité de succès. Ces indicateurs de performance permettent d'étudier d'une manière qualitative la stabilité du protocole et d'une manière quantitative le gain en débit et délai de transmission. Notre solution s'avère être prometteuse et plus performante que slotted aloha puisqu'elle assure un débit plus élevé et surtout un délai d'attente largement réduit (le délai augmente exponentiellement dans le cas de slotted aloha). Notre solution est donc robuste, efficace, facile à implémenter et pourrait rendre cette famille de protocoles capable de supporter les applications interactives et temps-réel.

## Bibliographie

1. Abramson N. The aloha system – another alternative for computer communications. *AFIPS Conference Proceedings*, Volume 36, Pages 295-298, 1970.
2. Altman E, Barman D, Benslimane A, and El-Azouzi R. Slotted aloha with priorities and random power. *Proceedings of IFIP Networking*, Ontario, Canada, Pages 610-622, 2005.
3. Altman E, El-Azouzi R, and Jimenez T. Slotted aloha as a Stochastic Game with Partial Information. *Proceedings of WiOpt'03*, Sophia Antipolis, France, March 3-5, 2003.
4. Bertsekas D, and Gallager R. *Data Networks*, PRENTICE HALL, Englewood Cliffs, New Jersey, 1987.
5. El-Azouzi R, Sabir E, Benarfa S, Jimenez T, and Bouyakhf E. H. Cooperative and noncooperative control for slotted aloha with random power level selection algorithms". *Proceedings of valuetools'07*, Nantes, France, 23-25 October, 2007.
6. LaMaire R.O, Krishna A, and Zorzi M. On the randomization of transmitter power levels to increase throughput in multiple access radio systems. *Wireless Networks*, Volume 4, Pages 263-277, 1998.
7. MacKenzie A. B, and Wicker S. B. Selfish users in aloha : A game theoretic approach". *Proceedings of 54th Vehicular Technology Conference (VTC'01)*, Atlantic City, NJ, Volume 3, Pages 1354-1357, 2001.
8. MacKenzie A. B, and Wicker S. B. Stability of slotted aloha with Multi-packet Reception and Selfish Users. *Proceedings of IEEE INFOCOM*, Volume 3, Pages 1583-1590, April, 2003.
9. Metzner J. On improving utilization in ALOHA networks. *IEEE Transaction on Communications*, Volume 24, Issue 4, Pages 447-448, 1976.
10. Park J, and van der Schaar M. Stackelberg Contention Games in Multi-User Networks. To appear in *EURASIP Journal on Advances in Signal Processing*, 2009.
11. Roberts L. G. Aloha packet system with and without slots and capture. *Tech. Rep. Ass Note 8*, Stanford Research Institute, Advance Research Projects Agency, Network Information Center, 1972.
12. Sabir E, El-Azouzi R, and Hayel Y. A hierarchical slotted aloha game. *Proceedings of IEEE/ICST International conference on game theory for networks (GameNets'09)*. Istanbul, Turkey, 13-15 May, 2009.