
Évaluation d'une méthode de prédiction des déplacements de terminaux dans les réseaux mobiles¹

Jean-Marc François — Guy Leduc — Sylvain Martin

Institut Montefiore, B28
Université de Liège
B-4000 Liège Sart-Tilman, Belgique
Tel. : +32-4-3662871
Fax. : +32-4-3662989
{francois,leduc,martin}@run.montefiore.ulg.ac.be

RÉSUMÉ. Dans les réseaux mobiles, la prédiction du déplacement des terminaux fait régulièrement l'objet d'études : c'est une étape importante sur le chemin des garanties de QoS dans ces réseaux. Dans les infrastructures actuelles, les sources d'informations permettant d'induire le déplacement d'un mobile sont multiples ; de plus, le cas des terminaux incapables d'émettre de telles informations est aussi fréquent qu'important. Dans ces conditions, pour qu'un mécanisme de prédiction puisse se déployer, il lui faut pouvoir s'adapter à cette diversité de façon flexible. L'évaluation d'un tel mécanisme est le sujet de cet article ; il s'agit d'une méthode simple et s'appuyant sur peu d'hypothèses.

ABSTRACT. In mobile networks, terminals mobility prediction is regularly studied : it's an important milestone towards QoS guarantees in those networks. With the current infrastructure, the available sources of information from which to deduce mobiles' trajectories are numerous; furthermore, the important particular case of mobiles unable to emit such information is both frequent and important. Under these circumstances, a prediction mechanism must be able to adapt to various conditions in a flexible way. The evaluation of such a mechanism is the subject of this paper; it is a simple method based on a small number of hypotheses.

MOTS-CLÉS : Mobilité, prédiction, modèle de Markov caché

KEYWORDS: Mobility, prediction, hidden Markov model

1. Ce travail a été financé par la Région Wallonne (DGTRE) dans le cadre du programme "Wallonie-Développement-Université" (projet ARTHUR) et par les Services fédéraux des affaires scientifiques, techniques et culturelles (SSTC, Belgique) dans le cadre des PAI (projet P5/11 MOTION).

1. Introduction

Les réseaux mobiles sont une réalité de tous les jours, et peu doutent du fait qu'ils se généralisent encore bien plus dans un futur proche.

Le problème de la garantie de QoS, qui n'a pas encore de réponse définitive dans les réseaux classiques, est encore plus complexe dans les réseaux mobiles : au cours de ses déplacements, un hôte mobile (MH) dialogue avec plusieurs points d'accès (AP) qui lui donnent sa connectivité. Le passage d'un AP au suivant (*handover*) est une cause de rupture de service : un protocole comme *mobileIP* impose dans ce cas une lourde phase signalant le changement de localisation¹. De plus, connaître la façon dont les mobiles se déplacent habituellement permet la mise en place de ressources (par exemple, la réservation de bande passante pour certaines classes de services).

Les performances peuvent donc être améliorées en prévoyant les handovers et en réalisant des enregistrements ou des réservations de ressources proactifs.

Le problème peut être abordé de deux façons : statistiquement ou au cas par cas.

L'approche statistique consiste à déduire les probabilités de déplacement d'un mobile du comportement habituel d'autres mobiles liés au même AP. Dans [HAD 99] et [HAD 02], cet apprentissage se fait grâce à un *learning automaton*. Dans [CHO 98], des statistiques plus précises sont dressées en tenant compte des temps de séjour des MHs, de leur provenance et de la période de la journée. Ces informations peuvent être utiles pour deviner par où un mobile particulier s'en va ([HAD 99] et [HAD 02]) ou pour dimensionner des ressources ([CHO 98] étudie la réservation de bande passante pour minimiser raisonnablement la probabilité de rupture de connexion lors de handovers).

Une autre approche est la prédiction de la trajectoire de chaque mobile individuellement. Pour cela, les MHs peuvent stocker les trajets qu'ils empruntent habituellement et les fournir à leur AP pour l'aider dans son travail de prédiction ; cette méthode demande donc d'utiliser la bande passante aérienne pour le transfert de ces informations et fait jouer aux mobiles un rôle actif ([LIU 98] se base partiellement sur ce principe). Une autre solution est de recueillir des informations émises par un MH pour en déduire son déplacement ; [LIU 98] utilise la puissance d'antenne reçue par un MH et les propriétés géométriques supposées de cellules ATM pour en inférer les futures cellules qui seront atteintes. Dans [SOH 03], l'hypothèse est faite que chaque MH est équipé d'un GPS et que tout AP connaît la cartographie de la région qui l'entoure. Cela permet de trouver des routes sur lesquelles se produisent les handoffs pour les anticiper.

Cet article présente une méthode de prédiction se voulant simple et générique : elle s'adapte aux différentes informations que peuvent donner les MHs sur leur dé-

1. Des protocoles de *micro-mobilité* ([RAM 00, CAM 00]) ou la version hiérarchique de *mobileIP* ([CAS 98]) améliorent grandement les choses, mais ne changent pas la problématique de la question.

placement (grâce aux HMMs, un outil d'intelligence artificielle simple) et dégénère en une approche statistique en leur absence². La section 2 de cet article explique la méthode et les moyens pour la mettre en œuvre. Vient ensuite une courte introduction aux HMMs. Enfin, le résultat de simulations est présenté.

2. Mobiles, prédiction et HMMs

2.1. Principes

Dans les réseaux mobiles sans fil, un certain nombre de *points d'accès* (APs) relaient des *hôtes mobiles* (MHs) au réseau filaire ou à d'autres mobiles *via* un *routeur d'accès* (AR). Nous supposons qu'un MH est connecté à un seul AP à la fois. Quand un MH voit sa connectivité se dégrader alors qu'une communication avec un autre AP devient possible, il quitte le premier pour le second (auprès duquel il *s'enregistre*) ; cette transition est appelée *handover* et cause généralement une perturbation des transmissions en cours. Cette perturbation pourrait être minimisée si, devinant l'arrivée du handover et vers quel AP le mobile se dirige, des ressources pouvaient être mises en place anticipativement.

De là l'idée de prédiction de trajectoire de mobile, avec un double but : prédire vers quel AR un MH particulier se dirige, et, aussi, analyser statistiquement en quelle proportion ils se dirigent vers tel ou tel AR (et, éventuellement, mesurer leur débit moyen, ou leur exigence de QoS).

Dans cet article, une double hypothèse est faite :

- un mobile se connectant à un nouvel AR est capable de lui donner l'adresse de l'AR qu'il vient de quitter ;
- un mobile peut, au cours de son déplacement, émettre régulièrement un vecteur donnant une indication sur son trajet.

Ce vecteur (nommé *observation*) peut se déduire d'informations liées au protocole de niveau 2 ou d'un appareil dédié (dans le meilleur des cas, un GPS). Lors de son parcours, un MH émet donc une ou plusieurs observations, formant une séquence ; elle se dénote $S = O_1, O_2, \dots, O_i$ (i est appelé longueur de S).

Les différents mobiles se cantonnent à un rôle assez passif : ils se contentent de communiquer l'adresse de l'AR qu'ils viennent de quitter lors de leur enregistrement et reportent régulièrement une observation ; en particulier, ils ne doivent pas retenir leurs trajets les plus courants. C'est donc l'AR avec lequel le mobile est en communication qui va jouer un rôle central : dans un premier temps, il va faire l'apprentissage des différents motifs de déplacement des MHs pour pouvoir, ensuite, prédire l'AR vers lequel un MH se rend.

2. Des exemples d'extensions à cette méthode sont donnés dans [FRA 03].

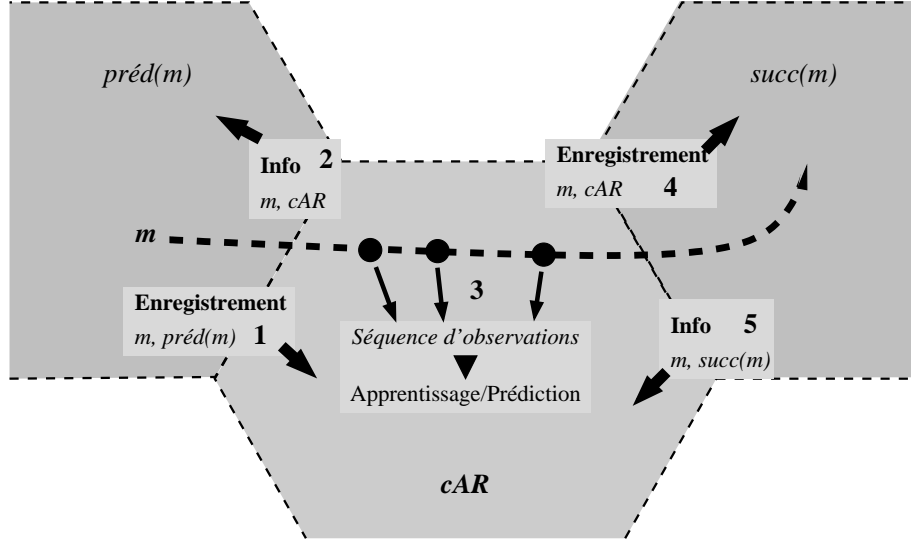


Figure 1. Vue d'ensemble des messages envoyés par les mobiles et ARs. Lors de chaque handover, les mobiles s'enregistrent auprès d'un nouvel AR (1, 4) ; à chaque fois, ils renseignent leur adresse et celle du routeur qu'ils viennent de quitter. Le routeur recevant cette information renseigne l'AR précédent de l'arrivée de ce mobile (2, 5). Lorsqu'il est enregistré, un mobile envoie une série d'observations (3) qui servira à un apprentissage (d'abord) ou au travail de prédiction de déplacement (ensuite).

Dans le reste de ce document, on dénotera $préd(m)$ et $succ(m)$ les ARs enregistrés avec un MH m respectivement avant et après le routeur qui fera la prédiction, cAR (voir fig. 1).

2.2. Plus en détail

Un routeur d'accès voulant mettre en place une politique d'apprentissage des déplacements de mobiles doit en observer un certain nombre afin de pouvoir les modéliser. Ce temps d'observation doit être mis à profit pour faire une liste des ARs voisins ; plus précisément, on dira de A qu'il est *voisin* de cAR si, durant la période d'apprentissage, on a observé un mobile m tel que $préd(m) = A$ ou $succ(m) = A$; soit V l'ensemble des voisins et AR_i ($\forall i = 1, \dots, n$) ses n éléments (on notera simplement i plutôt que AR_i là où le contexte le permet).

Ensuite, on pourra associer à chaque couple $AR_i, AR_j \in V$ un modèle des séquences d'observations générées par les mobiles m tels que $préd(m) = AR_i$ et $succ(m) = AR_j$ (soit $S_{i,j}$ l'ensemble des séquences de tels mobiles). Il a été choisi ici de modéliser ces séquences en s'appuyant sur des *modèles de Markov cachés* (HMMs).

Les HMMs modélisent des observations provenant de processus se trouvant à un instant t dans un état q_t et dont l'état à l'instant $t + 1$ ne dépend que de q_t (processus sans mémoire, ou "markoviens du premier ordre"). Dans le cas qui nous intéresse, ces états sont une image de la progression du mobile, progression qui nous est inconnue (on parle d'états *cachés*). On se persuade facilement que cette condition "d'amnésie" n'est pas bien respectée dans notre cas³, ce qui n'empêche pas la méthode de donner de bons résultats⁴.

Si les états sont cachés, ils sont néanmoins observables indirectement *via* les observations. A chaque état est donc associé une densité de probabilité permettant d'estimer, lorsqu'une observation survient, si il est vraisemblable qu'elle lui corresponde.

Ceci est décrit plus formellement dans la section suivante.

3. Modèles de Markov cachés

3.1. Définitions

Un HMM ([RAB 86, RAB 89]) modélise des séquences d'observations $S = O_1, O_2, O_3, \dots$, comme expliqué ci-dessus. Chaque observation O_i correspond à un temps $t = i$.

Un HMM est caractérisé par :

- N états : Q_1, Q_2, \dots, Q_N . On notera q_t l'état du HMM à l'instant t .
- N fonctions de distribution de probabilité définies sur l'ensemble des observations et correspondant aux N états :

$$b_i(O) = P[O \text{ en } t \mid q_t = Q_i] \quad (\forall i = 1, 2, \dots, N) \quad [1]$$

Soit $B = \{b_1, \dots, b_N\}$.

- une matrice de probabilité de transition entre états : $A = \{a_{ij}\}$.

$$a_{ij} = P[q_t = Q_j \mid q_{t-1} = Q_i] \quad (\forall i, j = 1, 2, \dots, N) \quad [2]$$

Avec les propriétés suivantes :

$$0 \leq a_{ij} \leq 1 \quad (\forall i, j = 1, 2, \dots, N) \quad [3]$$

$$\sum_{j=1}^N a_{ij} = 1 \quad (\forall i = 1, 2, \dots, N) \quad [4]$$

3. Par exemple, la plupart des véhicules arrivant dans un carrefour en croix ne changent pas de direction ; sachant qu'un mobile se trouve dans un carrefour, connaître sa position quelques instants plus tôt permet donc de deviner assez sûrement par où il s'en va.

4. En fait, les HMMs sont la plupart du temps utilisés pour modéliser des processus non strictement markoviens.

– la probabilité que Q_i soit état initial d’une séquence :

$$\pi_i = P[q_1 = Q_i] \quad (\forall i = 1, 2, \dots, N) \quad [5]$$

Un HMM est habituellement dénoté $\lambda = (A, B, \pi)$, ce qui rappelle les paramètres entrant en jeu.

La probabilité d’observer une séquence d’observations S et une séquence d’états $Q = q_1, q_2, q_3, \dots, q_T$ étant donné un modèle λ est simplement la probabilité que q_1 soit état initial d’une séquence (notée ci-dessous $\pi_{q_1} = \pi_i$ avec i t.q. $q_1 = Q_i$), multipliée par la probabilité d’observer O_1 en q_1 , multipliée par la probabilité de passer de l’état q_1 à l’état q_2, \dots :

$$P[S | Q, \lambda] P[Q | \lambda] = \pi_{q_1} b(O_1) a_{q_1 q_2} b(O_2) a_{q_2 q_3} \dots a_{q_{T-1} q_T} b(O_T)$$

Pour calculer la probabilité de S étant donné λ , il ne reste qu’à sommer cette probabilité pour toutes les séquences d’états possibles :

$$P[S | \lambda] = \sum_{\text{tout } Q} P[S | Q, \lambda] P[Q | \lambda] \quad [6]$$

Un autre problème intéressant est celui-ci : étant donné une séquence d’observations S , comment peut-on trouver la séquence d’états la plus probable lui correspondant ?

$$Q^* = \operatorname{argmax}_Q P[S, Q | \lambda] \quad [7]$$

Le calcul de (6) et (7) peut se faire grâce aux algorithmes exposés dans [RAB 89].

3.2. Apprentissage

La figure 2 donne un aperçu d’un algorithme permettant d’ajuster les paramètres d’un HMM $\lambda = (A, B, \pi)$ pour qu’il modélise un ensemble de séquences d’observations donné. La solution de ce problème dépend du critère choisi pour estimer à quel point le modèle correspond aux séquences (dans l’algorithme exposé, on maximise la probabilité des séquences sur la suite d’états optimale donnée par l’équation 7) et, une fois ce critère choisi, il n’existe le plus souvent pas de méthode analytique permettant de calculer une solution exactement (voir [RAB 89] et [JUA 90]). Néanmoins, on peut arriver efficacement à une réponse localement optimale à l’aide d’algorithmes itératifs.

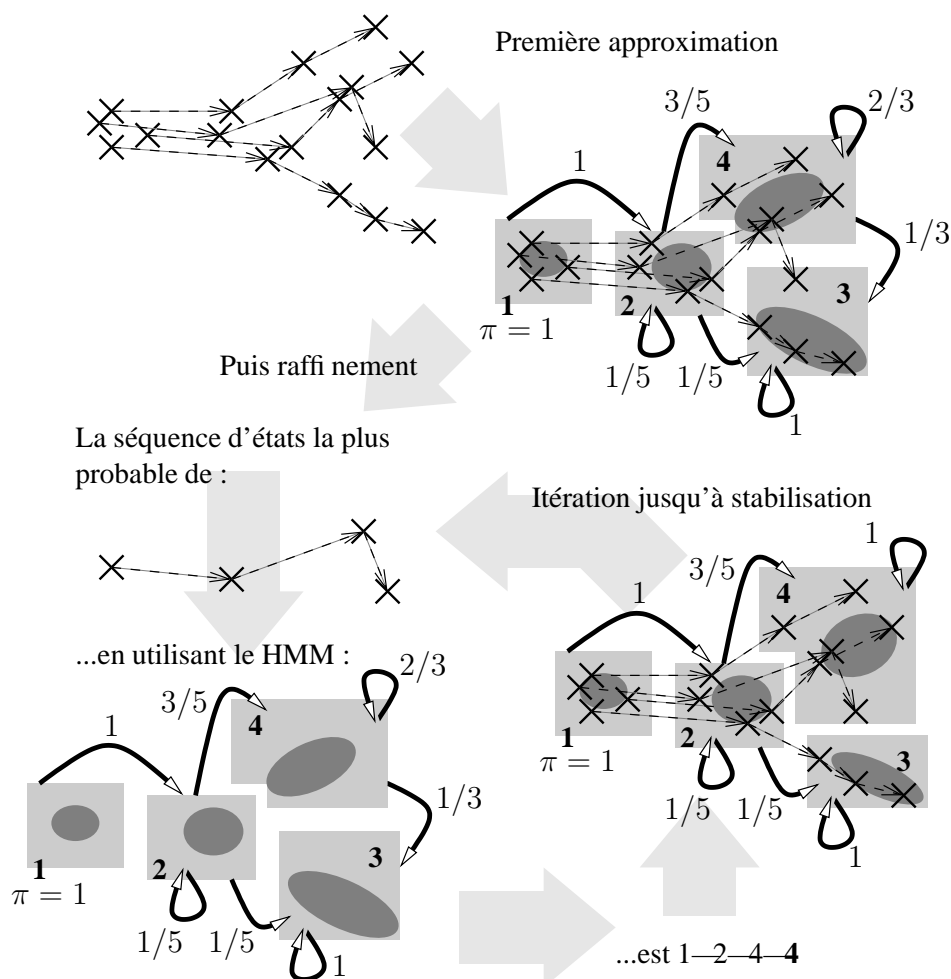


Figure 2. L'algorithme d'apprentissage *k-means* ([JUA 90]) : imaginons quatre voitures capables de délivrer régulièrement leurs coordonnées ; toutes viennent de l'ouest, trois partent vers le nord-est, une vers le sud-est (voir premier schéma). Une première approximation de HMM peut s'obtenir en groupant les observations les plus proches (deuxième schéma, rectangle clairs) ; à chacun de ces groupes correspond un état du HMM (on a choisi ici d'en utiliser 4). On en déduit : a) les poids des transitions entre états (directement tirés des séquences d'observations), et b) les fonctions de probabilité des observations, calculées en se servant des observations liées à l'état (ici représentées par l'ellipse d'erreur d'une probabilité gaussienne). Ensuite, on calcule, pour chaque séquence d'observations, la séquence d'états la plus probable associée (une seule séquence est représentée sur la figure) ; on peut alors comparer, pour chaque observation, son état le plus probable et l'état auquel il avait été associé à l'étape précédente. Si les deux ne concordent pas, le HMM est ré-estimé en effectuant la modification. Cette procédure est appliquée itérativement jusqu'à ce que plus aucune modification n'ait lieu. Notez que si la convergence est garantie quand les observations sont relatives à des variables discrètes, elle ne l'est pas dans le cas continu ; on doit alors faire des hypothèses additionnelles sur les fonctions b_i ([RAB 89, JUA 90]).

4. Mise en œuvre

4.1. Principes

Ce qui précède permet de munir assez simplement les ARs d'un mécanisme de prédiction basé sur les observations émises par les MHs. Sa mise en œuvre demande que chaque routeur observe des mobiles pendant un certain temps, puis :

- classe leur séquence d'observations entre les ensembles $S_{i,j}$ (c'est-à-dire l'ensemble des séquences des mobiles venant de AR_i et s'étant dirigés vers AR_j). Notons $n_{i,j}$ le nombre d'éléments de l'ensemble $S_{i,j}$;
- pour chacun de ces ensembles $S_{i,j}$, entraîne un HMM $\lambda_{i,j}$. Ceci peut se faire *via* l'algorithme expliqué figure 2.

Il est intéressant de prendre une période d'apprentissage assez courte pour permettre de s'adapter aux modifications des habitudes de déplacement au cours de la journée ; il faudra toutefois veiller à ce que le nombre de séquences d'observations soit assez grand pour permettre, à partir de cet échantillon, d'en généraliser un comportement type.

Le processus de prédiction peut avoir lieu dès que les HMMs ont été estimés. Chaque MH enregistré génère une série d'observations qui sont concaténées au fur et à mesure par l'AR pour former une séquence $S(m)$ dont la longueur augmente avec le temps. Chaque fois que $S(m)$ est modifié, on peut calculer sa probabilité $P[S | \lambda_{i,j}]$ de correspondre au HMM $\lambda_{i,j}$ grâce à l'équation 6. Reste à pondérer cette probabilité par la fréquence à laquelle les mobiles se dirigent vers AR_i préférentiellement aux autres voisins pour obtenir :

$$P_j = \frac{n_{préd(m),j}}{\sum_{i=1}^n n_{préd(m),i}} P[S | \lambda_{préd(m),j}] \quad \forall j = 1, \dots, n \quad [8]$$

...la probabilité que $succ(m) = AR_j$ pour un mobile m provenant de $préd(m)$.

Il faut donc laisser les séquences d'observations se construire jusqu'à constater qu'une probabilité P_j se dégage des autres P_i ($i \neq j$).

4.2. Passage à l'échelle

Plusieurs arguments permettent de conclure à la bonne tenue de la méthode présentée ici vis-à-vis du facteur d'échelle.

Du point de vue des échanges de données en fonction du nombre de mobiles et de cellules, il faut remarquer que les messages 1 et 4 (figure 1) sont courts (dans le cas de mobile IP, il s'agirait de deux adresses IP) et viennent s'ajouter à ceux, inévitables, du processus d'enregistrement (pour mobile IPv4, 116 bytes). Les messages 2 et 5 sont tout aussi courts et ne transitent que sur le réseau filaire. De leur côté, les messages contenant les observations peuvent rester peu nombreux (les simulations données ci-

après montrent que des séquences d'observations de longueur 10 donnent de bons résultats) et utiliser le mécanisme de *piggybacking*.

La charge de calcul imposée par l'apprentissage des modèles d'une part, et par le calcul de la probabilité d'une séquence étant donné un modèle d'autre part, reste modeste. En effet, ce premier calcul peut se faire *off-line* et ne nécessite que peu de temps si on se limite à faire l'apprentissage sur un échantillon de séquences représentatif (mais assez grand pour ne pas laisser le HMM apprendre un ensemble de cas particuliers⁵). De son côté, la probabilité d'une séquence pour un modèle donné se calcule en un temps proportionnel à N^2T (où N est le nombre d'états du modèle et T la longueur de la séquence, voir [RAB 89]); les probabilités en jeu ici peuvent donc se calculer rapidement⁶.

5. Simulations

5.1. Observations non bruitées

Il ne s'agit pas ici de trouver quelle distribution peut mieux qu'une autre modéliser tel ou tel genre d'observations, mais plutôt d'en trouver une qui puisse s'accomoder de tous les cas ; ceci explique que l'on se servira ici de fonctions de probabilités gaussiennes uniquement. Cela ajoute à la généricité de la méthode : aucune modification ne doit y être apportée pour l'adapter à un autre type d'observations ! Cela ne l'empêche pas de présenter de bonnes performances, qui pourront donc être augmentées en trouvant une distribution correspondant mieux aux observations utilisées.

Un des premiers facteurs pouvant influencer la bonne prédiction du prochain voisin est la complexité du déplacement d'un mobile dans la zone. Cela se comprend aisément : augmenter le nombre de chemins menant à une zone implique la construction de HMMs devant reconnaître chacune de ces trajectoires, donc des modèles plus complexes et une plus grande probabilité de voir deux chemins menant à des voisins différents générer des observations semblables.

Ainsi, plusieurs cartes de complexité croissante ont été réalisées ; sur chacune, des mobiles se déplacent à vitesse constante (mais aléatoire) le long des rues, dans le sens des flèches, et choisissent au hasard leur chemin quand un embranchement se présente. Les routes en contact avec le bord gauche des cartes sont celles par lesquelles les mobiles arrivent dans la zone.

Un second facteur pouvant influencer le bon fonctionnement de la méthode est la longueur moyenne des séquences d'observations utilisées. Intuitivement, des sé-

5. Une règle simple est d'utiliser un ensemble d'observations d'apprentissage dont la taille est beaucoup plus grande que le nombre d'états du HMM ; les simulations les plus complexes données plus loin nécessitent par exemple quelques centaines d'observations. Le programme Java écrit pour les exécuter fait ce travail en moins de deux secondes sur un ordinateur cadencé à 1.2GHz.

6. En maximum 3 ms dans le cadre des simulations de cet article.

quences longues donnent une idée plus précise de la trajectoire empruntée et donc facilitent le travail de prédiction. Dans les simulations présentées ci-après, les observations sont émises à intervalle régulier ; c’est la valeur de cet intervalle qui permet d’ajuster la longueur moyenne des séquences.

Un paramètre important doit être fixé : le nombre d’états composant les HMMs. Plus le nombre d’états est élevé, plus le modèle est précis. Bien sûr, le temps de calcul d’un HMM optimum pour des observations données augmente avec le nombre d’états, mais cette augmentation ne pose pas de problème dans le contexte qui nous intéresse : les modèles ne doivent être réévalués que de temps en temps, lorsque les habitudes de déplacement des mobiles ont varié.

La figure 3 donne les résultats des simulations effectuées.

Les conditions dans lesquelles a été fait l’apprentissage sont celles-ci : des mobiles sont envoyés dans les cartes dessinées figure 3. Pendant leur trajet, ils émettent des observations à intervalle de temps régulier. Une fois le MH arrivé à un bord de la carte, l’AR vers lequel il se dirige est déterminé et sa séquence d’observations est collectée. Cette séquence sert au travail d’apprentissage/prédiction comme expliqué ci-dessus.

La colonne “apprentissage” de la figure 3 donne les résultats d’un apprentissage basé sur la distance entre les mobiles et l’antenne avec laquelle ils communiquent. Cet apprentissage a été fait avec des HMMs de 5 et 10 états, et avec des séquences de longueur moyenne de 5 et 10 observations.

La colonne “statistique” donne les performances obtenues si on estime que les mobiles sont incapables d’émettre la moindre observation. Dans ce cas, le second facteur du membre de droite de la formule (8) vaut simplement 1, et la prédiction n’est plus que statistique. Si cette façon de faire peut sembler simpliste, elle n’en reste pas moins la seule utilisable dans le cas particulier fréquent de mobiles “aveugles”, tel un ordinateur portable ne disposant pas d’une interface sans fil. Il faut noter que ces statistiques peuvent être utiles pour la mise en place anticipative de ressources partagées (voir [CHO 98] ou [HAD 99]), et que, dans le *framework* présenté ici, elles sont accessibles gratuitement.

La formule (8) permet de classer les ARs voisins en une liste triée par ordre de probabilité décroissante. Les chiffres indiqués dans les colonnes “1er” renseignent le pourcentage de prédiction exacte, c’est-à-dire tel que $\text{succ}(m)$ est à la première place de cette liste. La colonne “2nd” donne la même information avec cette fois $\text{succ}(m)$ placé en première ou deuxième position.

La première carte est fort simple mais représente un cas de figure important, puisque de vastes zones susceptibles d’être couvertes par des réseaux mobiles sont peu peuplées (donc traversées par peu de routes et caractérisées par des motifs de déplacement simples et peu nombreux). Dans ces conditions simples, les performances sont quasiment parfaites. Les résultats statistiques se déduisent bien sûr de la symétrie de la cartographie mise en place.

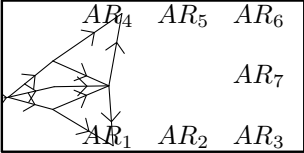
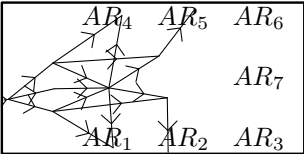
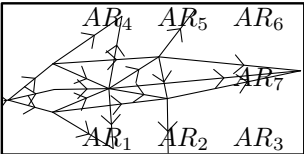
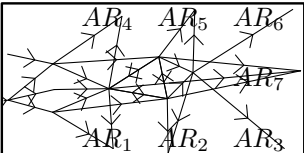
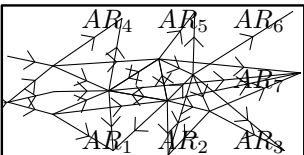
| CARTE | APPRENTISSAGE | | | | STATISTIQUE | |
|---|---------------|---------|------|----------|-------------|----------|
|  | Long. | 5 états | | 10 états | | 1er 2nd |
| | | 1er | 2nd | 1er | 2nd | 51% 100% |
| 5 | | 100% | 100% | 99% | 100% | |
| 10 | | 100% | 100% | 100% | 100% | |
|  | Long. | 5 états | | 10 états | | 1er 2nd |
| | | 1er | 2nd | 1er | 2nd | 26% 51% |
| 5 | | 89% | 98% | 97% | 99% | |
| 10 | | 94% | 100% | 95% | 100% | |
|  | Long. | 5 états | | 10 états | | 1er 2nd |
| | | 1er | 2nd | 1er | 2nd | 33% 55% |
| 5 | | 91% | 98% | 91% | 100% | |
| 10 | | 91% | 98% | 93% | 100% | |
|  | Long. | 5 états | | 10 états | | 1er 2nd |
| | | 1er | 2nd | 1er | 2nd | 30% 59% |
| 5 | | 67% | 88% | 78% | 91% | |
| 10 | | 79% | 93% | 81% | 95% | |
|  | Long. | 5 états | | 10 états | | 1er 2nd |
| | | 1er | 2nd | 1er | 2nd | 30% 57% |
| 5 | | 67% | 88% | 72% | 87% | |
| 10 | | 78% | 95% | 82% | 94% | |

Figure 3. Simulations avec des observations non bruitées. La colonne de gauche représente la carte utilisée pour la simulation. La colonne du milieu donne les résultats d'un apprentissage basé sur la distance des mobiles à l'AP, et celle de droite un apprentissage sans observation (purement statistique). Les colonnes désignées par "1er" donnent le pourcentage de prédiction exacte, celles désignées par "2nd" comportent le pourcentage de prédiction exacte, ou telle que la seconde meilleure prédiction était la bonne.

Les deux cartes suivantes font augmenter le nombre d'ARs voisins ; les performances restent dans tous les cas acceptables : on se rend compte de l'efficacité de modèles de Markov simples associés à des séquences courtes. D'un point de vue statistique, on se rend compte que AR_7 est le prochain routeur probable de la troisième carte ; cela facilite le travail de prédiction malgré l'augmentation du nombre de voisins possibles. Le modèle de déplacement choisi ici (purelement aléatoire, alors qu'il existe en réalité des routes principales plus probables que d'autres) peut être vu comme pessimiste : il est raisonnable de supposer que, dans beaucoup de situations réelles, ce phénomène de chemin fréquent est souvent présent.

Les deux dernières cartes rendent le problème de plus en plus complexe : les MHs disposent nouveau point d'entrée, ajout de nouvelles routes, de nouveaux voisins potentiels et introduction de chemins avec cycles. Les modèles les plus simples (peu d'états, courtes séquences) montrent leurs limites, alors que les plus précis affichent toujours de bons résultats. Entre ces deux extrêmes, on constate que des séquences assez longues sont indispensables pour pouvoir discerner au maximum les trajectoires possibles.

5.2. Influence de l'ajout de bruit

La carte la plus complexe de la figure 3 a été utilisée pour analyser l'influence de bruit sur les performances des HMMs.

Ce bruit est ajouté aux observations de façon agressive ; il est proportionnel à la valeur du signal mesuré x suivant la formule :

$$x_b = (1 + G_{0,1}f) x \quad [9]$$

où $G_{0,1}$ est tiré aléatoirement d'une distribution de probabilité gaussienne de moyenne $\mu = 0$ et de variance $\sigma^2 = 1^7$.

Des simulations ont été menées pour plusieurs facteurs f ; leurs résultats sont donnés figure 4.

5.3. Étude des résultats

Les résultats respectent bien l'intuition selon laquelle augmenter la longueur des séquences d'observations ou la complexité du modèle HMM ne peut qu'améliorer les choses.

Dans tous les cas, utiliser les HMMs donne des résultats systématiquement meilleurs que ceux obtenus par observation statistique : l'utilisation de HMM est toujours justifiée.

7. On ne s'attache pas à la modélisation du bruit obtenu pour un type de transmission particulier ; on l'exprime ici d'une façon raisonnable mais pessimiste.

| $f = 0\%$ | | | | | $f = 5\%$ | | | | |
|-----------|---------|-----|----------|-----|-----------|---------|-----|----------|-----|
| Long. | 5 états | | 10 états | | Long. | 5 états | | 10 états | |
| | 1er | 2nd | 1er | 2nd | | 1er | 2nd | 1er | 2nd |
| 5 | 67% | 88% | 72% | 87% | 5 | 64% | 88% | 69% | 86% |
| 10 | 78% | 95% | 82% | 94% | 10 | 71% | 92% | 71% | 93% |

| $f = 10\%$ | | | | | $f = 15\%$ | | | | |
|------------|---------|-----|----------|-----|------------|---------|-----|----------|-----|
| Long. | 5 états | | 10 états | | Long. | 5 états | | 10 états | |
| | 1er | 2nd | 1er | 2nd | | 1er | 2nd | 1er | 2nd |
| 5 | 58% | 83% | 61% | 81% | 5 | 50% | 75% | 54% | 75% |
| 10 | 59% | 87% | 64% | 87% | 10 | 51% | 81% | 53% | 83% |

Figure 4. Simulations avec des observations bruitées en fonction du facteur de bruit f (voir équation 9). Ces résultats sont à mettre en relation avec ceux de la figure 3.

N'avoir à sa disposition que de courtes séquences ne doit pas faire penser que la méthode ne peut mener qu'à des performances marginales : dans les simulations faites ici, l'utilisation de séquences courtes n'entraîne, dans le pire des cas, qu'une différence de résultats de 15%.

À longueurs de séquences d'observations égales, augmenter le nombre d'états des HMMs entrant en jeu est conseillé. Symétriquement, se servir de séquences longues avec des HMMs simples (en cas de ressources en calcul et/ou mémoire limitées) a un intérêt.

L'ajout de bruit fait baisser les performances : dans le cas de cartes complexes et avec les observations étudiées ici, certaines trajectoires deviennent indiscernables. Pour obtenir un résultat exploitable, il faudra utiliser les deux voisins les plus probables plutôt qu'un seul.

6. Conclusions

Cet article présente une méthode de prédiction de mobilité pour réseaux mobiles assez générique pour s'adapter de façon transparente aux différentes informations émises par les mobiles pouvant en bénéficier. L'utilisation d'une méthode d'intelligence artificielle simple lui permet de s'affranchir de nombreuses hypothèses (par exemple, la géométrie des cellules traversées peut être quelconque).

Elle n'implique que les ARs et peut donc se déployer petit à petit, laissant aux MHs un rôle minimal.

Des MHs incapables d'émettre des informations de déplacement peuvent tout de même être utilisés pour une mesure statistique des trajectoires les plus fréquentes (ceci

peut être particulièrement utile dans les réseaux mobiles ne reposant pas sur une technologie sans fil).

Les performances de cette méthode ont été quantifiées à l'aide de différentes simulations.

La mise en œuvre des méthodes de prédiction reste un sujet ouvert aux investigations : quelles ressources peut-on mettre en place anticipativement ? Comment le faire ? Quel bénéfice en tire-t-on ? A quel moment une approche statistique montre-t-elle ses limites ? Ces questions n'ont pas encore de réponse définitive et devront être étudiées dans l'avenir.

7. Bibliographie

- [CAM 00] CAMBELL A. T., GOMEZ J. et al., « Cellular IP », Internet draft, January 2000, draft-ietf-mobileip-cellularip-00.txt.
- [CAS 98] CASTELLUCCIA C., « A Hierarchical Mobile IPv6 Proposal », Technical report, November 1998, INRIA TR-226, <http://www.inrialpes.fr/Planete/people/ccastel/>.
- [CHO 98] CHOI S., SHIN K. G., « Predictive and adaptive bandwidth reservation for hand-offs in QoS-sensitive cellular networks », *ACM SIGCOMM Computer Communication Review*, vol. 28, n° 4, 1998, p. 155–166, ACM Press.
- [FRA 03] FRANÇOIS J.-M., LEDUC G., MARTIN S., « Learning movement patterns in mobile networks : a generic method », Internal report, April 2003.
- [HAD 99] HADJIEFTHYMIADES S., MERAKOS L., « ESW4 : enhanced scheme for WWW computing in wireless communication environments », *ACM SIGCOMM Computer Communication Review*, vol. 29, n° 5, 1999, p. 24–35, ACM Press.
- [HAD 02] HADJIEFTHYMIADES S., MERAKOS L., « Using Path Prediction to Improve TCP Performance in Wireless/Mobile Communications », *IEEE Communications Magazine*, vol. 40, n° 8, 2002, p. 54–61.
- [JUA 90] JUANG B. H., RABINER L. H., « The segmental k-means algorithm for estimating the parameters of hidden Markov models », *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 38, n° 9, 1990, p. 1639–1641.
- [LIU 98] LIU T., BAHL P., CHLAMTAC I., « Mobility Modeling, Location Tracking, and Trajectory Prediction in Wireless ATM Networks », *IEEE JSAC*, vol. 16, n° 6, 1998, p. 922–936.
- [RAB 86] RABINER L. R., JUANG B. H., « An introduction to hidden Markov models », *IEEE ASSP Magazine*, vol. 34-3, 1986, p. 4–16.
- [RAB 89] RABINER L. R., « An tutorial on hidden Markov models and selected applications in speech recognition », *Proceedings of the IEEE*, vol. 77, n° 2, 1989, p. 257–282.
- [RAM 00] RAMJEE R., LA PORTA T., THUEL S. et al., « IP micro-mobility support using HAWAII », Internet draft, July 2000, draft-ietf-mobileip-hawaii-01.txt.
- [SOH 03] SOH W.-S., KIM H. S., « QoS Provisioning in Cellular Networks Based on Mobility Prediction Techniques », *IEEE Communications Magazine*, vol. 41, n° 1, 2003, p. 86–92.