

# Myopic versus Clairvoyant Admission Policies in Wireless Networks

Pierre Coucheney  
INRIA  
51 av. J Kuntzmann,  
38330 Montbonnot, France  
pierre.coucheney@inria.fr

Emmanuel Hyon  
LIP6/Univ. Paris Ouest Nanterre  
104 av. du Président Kennedy  
75016 Paris, France  
emmanuel.hyon@u-paris10.fr

Corinne Touati  
INRIA  
51 av. J Kuntzmann,  
38330 Montbonnot, France  
corinne.touati@inria.fr

Bruno Gaujal  
INRIA  
51 av. J Kuntzmann,  
38330 Montbonnot, France  
bruno.gaujal@inria.fr

## ABSTRACT

In this paper, we consider a dynamic scenario in which mobile users with elastic traffic arrive at a wireless heterogeneous system according to a Poisson arrival process. The wireless system consists of a set of overlapping network cells of different technologies. The protocols associated to each cell specify the throughput allocated to each user, given the cell load (i.e. the number of active users and their geographical positions). Meanwhile, mobile users (the players) can choose which cell to associate to. Suppose they collaborate to obtain an efficient and fair share of the global throughput. Then, two families of mechanisms can be considered: (i) either the users bargain at each stage of the system so as to obtain, given the present population, a fair share of the present throughput (repeated single-stage games), or (ii) users bargain so as to obtain a fair *expected* throughput during the *total* duration of their call (single stochastic game), taking into account the knowledge of the global (stochastic) arrival process and connection size. In this paper, we numerically compare the two policies and their resulting performance: does forecasting the future significantly impact the optimal mobile-to-cell association?

## Categories and Subject Descriptors

C.2 [Computer-communication Networks]: Network Operations, Internetworking.; G.3 [Probability and Statistics]: Stochastic processes.

## General Terms

Performance.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. GAMECOMM 2009, October 23, Pisa, Italy  
Copyright © 2009 ICST 978-963-9799-70-7  
DOI 10.4108/ICST.VALUETOOLS2009.7766

## Keywords

Game Theory, Network Management, Markov Decision Processes.

## 1. INTRODUCTION

With the increasing level of wireless internetworking, optimal association of mobiles to the most appropriate cell has become a central challenging problem. On the one hand, network providers now offer their services through a variety of different technologies, including UMTS, WiMAX, WiFi, LTE and others. On the other hand, recent mobile devices are compliant with several technologies simultaneously. Then, given a set of overlapping cells of various technologies and a set of mobile users such that each of them can connect to a subset of these cells, a natural question is of how to map mobile users to these cells so as to optimize their quality-of-service (QoS).

A few recent articles have tackled this question. Among them, two families can be observed. On the first one, the optimisation is made on the fly, only considering the present state of the system (see, e.g. the distributed iterative algorithm of [5]). On the second one, the arrival process and the call duration are known stochastically and impact the association decision. Such optimizations are generally studied by SMDP (Semi Markov Decision Process) approaches to solve the problem of maximizing the global expected throughput of the system by a central controller in a dynamic scenario (see [6, 10]). The former approach aims at developing algorithms to be implemented in mobile devices, while the latter has been designed for performance evaluation, network dimensioning and provisioning purposes.

From a game theoretic point of view, the approach [5] is most typical of multiple *single-stage* (or static) games. Indeed, at each stage (arrival or departure of a mobile), the game is reset and the outcome of the previous one is forgotten: the optimization does not take into account the history of the system. On the other hand, the approach of [6, 10] closely relates to a particular class of repeated games called *stochastic games*, as by Lloyd Shapley [12]. Such games consist of a sequence of stages at which the game is played (dynamic games) and where the players select actions (choose network cells) so as to optimize their average

utility (throughput) defined as the sum of the stage payoffs. Further, these games have probabilistic transitions between the stages and are therefore naturally studied by means of a SMDP.

In this paper, first we formulate the optimal mobile-to-cell association problem under both scenarios (with and without considering the stochastic arrival process) under the game theoretic framework. We call the optimum point in the multiple single-stage game the *nearsighted optimum* (or *myopic game*) and on the stochastic game the *longsighted optimum* (or *clairvoyant optimum*). Looking at the optimal mobile-to-cell association problem from a game theoretic framework point of view opens the way for future works both on the performance evaluation point of view and the design on new distributed algorithms. Second, we solve both problems numerically and compare them through extensive simulations. We investigate the impact of the benefits of longsighted approach with respects of the different system parameters (such as the arrival rate and demand).

The rest of the paper is organised as follows. We first introduce the model and notations used throughout the paper (Section 2). We then present the two models starting from the nearsighted game (Section 3) before moving on to the longsighted game (Section 4). Finally, Section 5 and 6 present the different numerical scenarios we considered and our numerical results along with their analysis, respectively.

## 2. MODEL

In this section, we present the model and the optimization problems we studied. We first introduce the physical system of wireless heterogeneous network (Subsection 2.1). We then move on to the description of the two kind of games considered (Subsection 2.2) before finally introducing the corresponding objective function for the two games (Subsection 2.3).

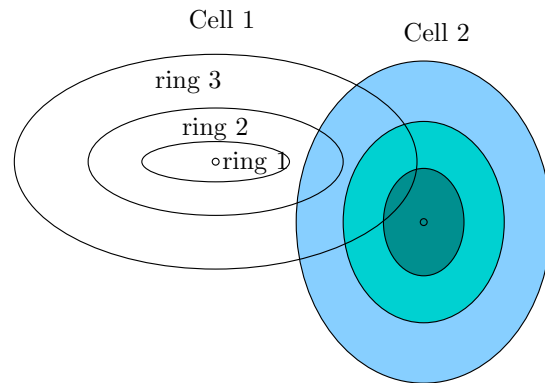
### 2.1 Overall Physical System

#### 2.1.1 Heterogeneous Wireless Networks

Consider a set of overlapping cells of different technologies, as illustrated in Figure 1. The geographical area covered by each cell is discretised in *rings* of identical characteristics (in terms of interference and attenuation levels). Then, all mobiles in the same ring are assumed to receive the same throughput.

For each cell  $c$ , let us denote by  $r_c$  the number of rings and  $n_c^j$  the number of (active) mobiles present in ring  $j$  of cell  $c$ . Without loss of generality<sup>1</sup>, we can assume that the number of rings in all the cells is the same:  $\forall c, r_c = r$ . The geometrical intersection of rings define *zones* (for instance, in Figure 1, there are 10 zones partitioning the two cells). One should note however that two mobiles in the same ring of a given cell receive the same bandwidth whatever their zones. Zones will only be useful for mobiles to decide to which cells they can connect. Hence, the load of cell  $c$  in terms of number of mobiles can be represented by the vector  $\mathbf{n}_c = (n_c^1, n_c^2, \dots, n_c^r)$ . We suppose that cells associated to two different technologies do not interfere. Hence, the service rate (or bandwidth) of a user connected to cell  $c$  is a function of vector  $\mathbf{n}_c$ . For each cell  $c$  and load  $\mathbf{n}_c$ , we de-

<sup>1</sup>Indeed, we can consider  $r = \max_c r_c$  and then to each cell  $c$ , add  $r - r_c$  dummy zones with load 0.



**Figure 1: A Heterogeneous Wireless System consisting of 2 cells with 3 rings each, defining 10 zones (by intersections).**

note by  $d_c^i(\mathbf{n}_c)$  the service rate received by a mobile in zone  $i$  and by  $\mathbf{n}$  the vector of the global distribution of customers  $\mathbf{n} = (\mathbf{n}_{c_1}, \dots)$ . Finally, for each active mobile  $p$ , we denote by  $c(p)$  its cell and by  $i(p)$  its ring.

Independently of the game at hand, the system guarantees an instantaneous minimum QoS [9]: each active mobile in the system should receive a service at least equal to  $d_{\min}$ . This amounts to say that the system has a finite capacity in terms of numbers of mobiles and the admission policy consists in accepting or refusing new calls from mobiles. Indeed, in practical cases, the network operators usually choose to favour active calls from new calls, and prefer blocking a new call than interrupting an existing one.

#### 2.1.2 Arrival Process and User Demands

We assume that mobiles join the whole system according to a Poisson process of parameter  $\lambda$ . The arrivals occur in a dedicated zone  $k$  with probability  $p_k$ . Therefore, arrivals in zone  $k$  follow a Poisson process with rate  $\lambda p_k$ . We further assume that the traffic is elastic, *i.e.* each mobile call is characterized by its size (as opposed to its duration for real-time traffic). We suppose that the user call sizes follow an i.i.d. exponential distribution with parameter  $\mu$ . Finally, we suppose the mobiles stay still during the whole duration of their communication (no mobility). So, we do not consider transition in a same cell from one ring or zone to another.

#### System behavior and admission control.

The system behaves in continuous time and its *state* only changes when an event occur. These events are arrivals or departures from the system. In case of arrival, a decision concerning the admission of the new mobile must be taken (should the call be accepted, and if so, which cell should carry the call when several choices are possible). Accepting a mobile changes the state of the system, which will last until a new event occurs. We trigger of the actions (issued from decisions) only when transitions occur and not at any time. However, since transitions are exponentially distributed and immediate payoffs do not depend on the duration of the transition, from [11] this does not degrade the solution of the decision problem.

## A Markovian system.

We now summarize the conditions under which the system keeps its markovian properties. The competing access model within each cell should be of *Generalized Processor Sharing* type [4], in which the resources are shared under a state dependant processor sharing policy. It has to be *work-conservative* [8], that is the total amount of work needed for each mobile should not be modified by the policy (but the total service time could). Under these conditions, the system is Markovian with arrival rate  $\lambda p_k$  (before admission) in zone  $k$  and has a state dependent departure rate  $\mu(\mathbf{n}_c) = \mu d_c^i(\mathbf{n}_c)$  in ring  $i$  of cell  $c$ .

## 2.2 Game Theoretic Models

In all models we suppose that the mobiles are *not* aware of their size. Hence, their estimation of their actual sojourn time only depends on their transmission rate  $d$ .

### 2.2.1 Single-Stage or Shortsighted Games

In game theory, a single-stage game is defined as a triplet  $(\mathcal{N}, \mathcal{C}, \mathcal{S})$  where  $\mathcal{N}$  is the set of players (here the mobile users),  $\mathcal{C}$  the set of costs. Here, we consider as a cost the sojourn time of an accepted mobile:

$$\forall p, c_p^{SS} = \begin{cases} \frac{1}{\mu} \cdot \frac{1}{d_{c(p)}^{i(p)}} & \text{if } p \text{ is accepted,} \\ K & \text{otherwise.} \end{cases} \quad (1)$$

$K \geq 0$  represents the unhappiness for a mobile to be rejected from the system. Finally,  $\mathcal{S}$  is the strategy space, which is here a vector of size  $|\mathcal{N}|$  so that each  $s_p$  with  $p \in \mathcal{N}$  is the set of cells that user  $p$  can connect to. This game can be referred as a *myopic* or *shortsighted*, as the utility and the decision only take into account the present time. As time goes on and users arrive and leave the system, the game is repeated: this model is a superposition of multiple *single-stage games*.

### 2.2.2 Stochastic or Longsighted Games

A stochastic game is a type of repeated game, with a infinite number of stages (time epoch) and a finite number of states  $M$  (mapping of active users to network cells). At the end of each stage the game is in some state. The game then moves to a new random state whose distribution  $P$  depends on the previous state, the actions chosen by the players (the association scheme, which translate into a departure rate) and an exogeneous stochastic process (the arrival process). The players select actions and each player receives a payoff (throughput) that depends on the current state and the chosen actions. The play goes on for an infinite number of stages. The total payoff to a player is taken either to be the discounted sum of the stage payoffs or the limit inferior of the averages of the stage payoffs.

As for the shortsighted game, we consider as a cost function, the sojourn time experienced by a mobile. In classical stochastic games, the user average cost function  $c_p^{LS}$  given its instantaneous cost  $c_p^{SS}$  can be written:

$$c_p^{LS} = \begin{cases} \gamma \int_{t=0}^{\infty} (1-\gamma)^{t-1} c_p^{SS}(t) \cdot dt & \text{in the } \textit{discounted game} \text{ with} \\ & \text{discount factor } \gamma \text{ (} 0 < \gamma \leq 1 \text{).} \\ \lim_{T \rightarrow \infty} \frac{1}{T} \int_{t=0}^T c_p^{SS} \cdot dt & \text{in the } \textit{undiscounted game}. \end{cases}$$

Both kind of  $n$ -player games can be studied with  $n$  superposed dynamical tools, e.g. *competitive MDP* [7]. Yet, the discounted game gives priority to mobiles arriving early in the system, as the discounted factor would then be low. Hence, we focus, in the remaining of this paper, on the undiscounted game. Also, as the mobiles are present only during a limited time, the integral is actually finite. Denote by  $T_p$  the interval of time in which user  $p$  is present. Then, the user payoff in our longsighted game is:

$$c_p^{LS} = \begin{cases} K & \text{if } p \text{ is rejected,} \\ \frac{1}{\mu} \cdot \int_{T_p} \frac{1}{d_{c(p)}^{i(p)}(t)} \cdot dt & \text{otherwise.} \end{cases}$$

## 2.3 Optimization Problem

Given the choice of the game, we consider the *social optimum*, defined as the sum of the costs over all users. The social optimum can be, in certain cases, computed through a distributed optimization, either using fictitious prices (see, e.g. [1]) or by designing a game so that the Nash Equilibrium of the game corresponds with the social optimum of the original game [5].

For the shortsighted game, the optimization problem is (at each time)

$$\min \sum_{p \in \mathcal{N}} c_p^{SS},$$

or equivalently

$$\min \frac{1}{|\mathcal{N}|} \sum_{p \in \mathcal{N}} c_p^{SS}. \quad (2)$$

For the longsighted game, as opposed to traditional stochastic games, the number of players grows to infinity (while, at a given instant, there is only a finite number of players in the system). Hence, the optimization problem is:

$$\min \lim_{|\mathcal{P}| \rightarrow \infty} \frac{1}{|\mathcal{P}|} \sum_{p \in \mathcal{P}} c_p^{LS}. \quad (3)$$

The whole system we study consists of a dynamical process, as described in Section 2.1.2 for which a policy (either shortsighted or longsighted) is performed. Let  $c_p^{SS}(\pi)$  be the instantaneous cost of policy  $\pi$  defined as the value of Eq. 1 for policy  $\pi$ . Further, let  $\mathcal{P}_t$  be the set of active mobile users at time  $t$  (either arriving at the system or presently connected to a cell). Then, the performance of policy  $\pi$  can be written

$$C_\pi = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{t=0}^T \sum_{p \in \mathcal{P}_t} c_p^{SS}(\pi) dt. \quad (4)$$

In the following two sections, we propose several policies of interest. They vary according to the set of constraints applied on the mobile equipment and to the myopic or clairvoyant policies considered.

## 3. OPTIMAL POLICIES FOR THE SHORT-SIGHTED GAME

In this section, we study the outcome of the game in the case where players only consider the present state of the system, without knowledge or hypothesis about the future.

Two scenarios can be considered. On the first one (Section 3.1), once connected to a network, the mobiles need to remain on this network until their connection ends. (This

is the case for actual mobiles available nowadays: vertical handover cannot be performed while maintaining the connection.) Meanwhile, resent research is performed to design protocols allowing mobiles to switch from a network of one technology to another within a call, referred to as *soft vertical handover*. We study this scenario on Section 3.2.

### 3.1 Without Soft Vertical Handover (SVH) Capabilities

Suppose that mobiles must remain in the same network during the total duration of their communication. Then, a mobile leaving the system does not affect the strategies of the others, but only their instantaneous utility (throughputs).

Similarly, a new arrival does not affect the strategies followed by the mobiles already in the system. The strategy of the new mobile  $p$  differs depending whether a cooperative or a selfish game is considered. In both cases, recall the set of strategies (network cells) available to  $p$  is  $S_p$ . Among these cells, consider  $S'_p$  the set of cells such that if mobile  $p$  connects to, then all present mobiles receive a throughput higher than their minimum requirements  $d_{\min}$ . Then, at each stage, the game is played as follows:

---

#### Algorithm 1 Admission Control and Association Policies for the Myopic Game without SVH (*MnSVH*)

---

1. Let  $C^{\text{SS}}$  the actual value of the optimization of Eq. 2.
  2. If  $S'_p = \emptyset$  the newcomer cannot enter the game and  $C^{\text{SS}}$  is increased by  $K$ .
  3. Else, the newcomer joins the cell in this restricted strategy set  $S'_p$  that:
    - minimizes the total system sojourn time (collaborative policy, called *MnSVH-coop* in the experiments, Section 6).
    - minimizes the newcomer sojourn time (selfish policy, called *MnSVH-self* in the experiments, Section 6).
  4. Let  $C^{\text{SS}}_{\text{accept}}$  the value of the optimization of Eq. 2 with newcomer  $p$  in the system. If  $C^{\text{SS}}_{\text{accept}} < C^{\text{SS}} + K$  then the newcomer is accepted, otherwise the admission controller rejects its call and  $C^{\text{SS}}$  is increased by  $K$ .
- 

### 3.2 With Soft Vertical Handover (SVH) Capabilities

Now, suppose that soft vertical handover is implemented. This infers that at each stage of the game (arrival or departure), the previous association scheme is forgotten, and that the game is reset. As for the case with no SVH, new mobiles are accepted in the system only if there exists an association scheme such that *all* mobiles receive more than their minimal throughput  $d_{\min}$  and if the entrance of the mobile does not degrades the overall performance of more than  $K$ . The admission control and association scheme then follow Algorithm 2. When a mobile leaves the system, step 3 of the algorithm is performed (no call admission mechanism).

In the case of simple systems, the optimal and fair allocation can be computed using greedy algorithms. Indeed, if  $|S_p|$  is the number of strategies of mobile  $p$  (i.e. the

---

#### Algorithm 2 Admission Control and Association Policies for the Myopic Game with SVH (*MSVH-coop*)

---

1. Let  $C^{\text{SS}}$  the actual value of the optimization of Eq. 2.
  2. If there exists no mapping of the mobiles (including the newcomer) such that  $\forall p, d_p \geq d_{\min}$ , then the newcomer cannot enter the game and  $C^{\text{SS}}$  is increased by  $K$ .
  3. Else, the game is played so as to minimise Eq. 2. Let  $C^{\text{SS}}_{\text{accept}}$  the corresponding optimization value.
  4. If  $C^{\text{SS}}_{\text{accept}} < C^{\text{SS}} + K$  then the newcomer is accepted, otherwise the admission controller rejects its call and  $C^{\text{SS}}$  is increased by  $K$ .
- 

number of networks it can connect to), then the number of possible association schemes is  $\prod_{p \in \mathcal{N}} |S_p|$  at each stage of the system. However, as the system grows, such complexity clearly becomes prohibitive. To overcome such, we use the distributed algorithm proposed in [5] which is based a Nash equilibrium learning mechanism based on evolutionary game theory. The algorithm can be performed on a modified version of the game so that the Nash equilibrium of the new game corresponds to the social optimum solution of the original game.

## 4. COMPUTING THE OPTIMAL POLICY WITH FULL INFORMATION

Here, we consider that statistical informations (arrival distribution as well as the distribution of the size of the connections) are known to all mobiles as well as the current state of the whole system. Therefore, one can rather consider the system as a whole and address the optimization problem as a Semi Markov Decision Problem. Indeed, all the foregoing assumptions are compatible with the Semi Markov Decision Process (SMDP) framework and we will be using the associated results presented for example in [11] to solve it numerically. We detail the model for systems with no soft handover capabilities, although a similar approach could be use for the soft handover case.

### 4.1 SMDP model

This part is devoted to the presentation to the basic elements of the construction of the decision process.

#### 4.1.1 State space

Two kinds of events have to be taken into account in the problem since the actions permitted to the controller differ according to the type of event that occurred. Either it is a departure and nothing can be done or it is an arrival and the system has to decide whether the customer is admitted and on what cell, or whether it is rejected. So we enlarge the state space by adding to the number of customers a last component representing the current event. This component is  $k = 0$  if it is a departure or  $k$  in case of an arrival in zone  $k$ . We therefore denote by  $x$  the state of the SMDP with  $x = (\mathbf{n}, k)$ . The state space is denoted by  $X$ .

#### 4.1.2 Action space

We record the action of the controller by  $q$ . There is only three possible actions when a customer arrives in the system:

rejection (denoted by  $q = 0$ ) and acceptance. In this case  $q$  takes the value  $c$ , according to the cell  $c$  in which the customer is admitted, among all cells intersecting its arrival zone  $k$ . Instead, in case of departure from the system, there is no possible action left to the controller and value of  $q$  is not relevant. At last, since the cells have finite capacities it may happen that an arriving customer sees a (partially) full system. In this case, the number of actions allowed to the controller is restricted. The acceptance is only possible in the subset of possible cells which are not full, and, if the whole system is full, rejection is performed.

#### 4.1.3 Transition probabilities and transition time

We denote by  $\mathbb{P}(y|(x, q))$  the probability to switch to state  $y$  when the system is in state  $x = (\mathbf{n}, k)$  and action  $q$  is performed. We denote by  $\Lambda(x, q)$  the rate of the exponential distribution of the transition time when we are in state  $x$  and decision  $q$  is taken. We present now how to compute these two values.

First,  $\Lambda(x, q) = \lambda + \sum_{c,i} \mu d_c^i(\mathbf{n}_c)$  is the finite rate of event occurrence. In the following, we will uniformize the continuous process by  $\Lambda_{max} = \max_{q,x \in X} \Lambda(x, q)$ , to go back to discrete time.

As for the transition matrix, let  $e_c^i$  be the vector whose entries are all null except for a unit entry corresponding to ring  $i$  in cell  $c$ .

If the initial state is  $x = (\mathbf{n}, 0)$ , then the next event will be a departure and no action is taken. Therefore, the next state is

- $y = (\mathbf{n} - e_c^i, 0)$  with probability  $\mu d_c^i(\mathbf{n}_c)/\Lambda_{max}$ ;
- $y = (\mathbf{n}, k)$  with probability equal to  $p_k \lambda(\mathbf{n}_c)/\Lambda_{max}$  or
- $y = x$  with probability  $(\Lambda_{max} - (\Lambda(x, q)))/\Lambda_{max}$ .

If the initial state is  $x = (\mathbf{n}, k)$ , this means that the next event is an arrival in zone  $k$ .

- Under action  $q = c$  (meaning that the new mobile is allocated to cell  $c$  (its ring being  $i$ )), the next state is
  - $y = (\mathbf{n} + e_c^i, 0)$  with probability  $(\Lambda(x, q) - \lambda)/\Lambda_{max}$  or;
  - $y = (\mathbf{n} + e_c^i, k)$  with probability  $p_k \lambda/\Lambda_{max}$ , and
  - $y = x$  with probability  $(\Lambda_{max} - (\Lambda(x, q)))/\Lambda_{max}$ .
- Under action  $q = 0$  (meaning that the new mobile is rejected, the next state is
  - $y = (\mathbf{n}, 0)$  with probability  $(\Lambda(x, q) - \lambda)/\Lambda_{max}$ ;
  - $y = (\mathbf{n}, k)$  with probability  $p_k \lambda/\Lambda_{max}$  or
  - $y = x$  with probability  $(\Lambda_{max} - (\Lambda(x, q)))/\Lambda_{max}$ .

## 4.2 Immediate Costs

We detail now the immediate cost induced from a decision  $q$  in state  $x$ . These rewards have two components: the cost of the action and a cost related to the system state. The cost of an action is denoted by  $K$  and is non null only when  $q$  is a rejection. On the other hand, the cost due to state  $x$  is the total number of mobile being active during the current time slot:

$$c(x, q) = \mathbf{n}\mathbf{1} + K_q.$$

This instantaneous cost is chosen so that the total average cost is the same as for the game formulation given in Section 2.2 (see upcoming Equation 7).

## 4.3 Decisions

We call policy a sequence of decision rules  $\pi = (d_0, d_1, \dots)$ , in which each decision rule is a mapping from the information set to the action space. The most general set of policies is history-dependent randomized policies,  $\Pi^{HR}$ , but here mainly since the costs are bounded we can use classical results on infinite-horizon, time-homogeneous Markovian policies to state that *Markov Deterministic Stationary Policies* are a dominating set (denoted by  $\Pi$ ) (see [11]).

Here, we study a problem with average cost per stage. Thus, the value function of a policy  $\pi$  is then the average cost per stage is given by

$$J_\pi(x_0) = \lim_{N \rightarrow \infty} \frac{1}{N} \mathbb{E} \left( \sum_{k=0}^{N-1} c(x_k, q_k) \right) \quad (5)$$

where  $x_k$  is the state and  $q_k$  the decision at epoch  $T_k$  while  $T_k$  is the instant of the  $k^{th}$  point of a Poisson process with rate  $\Lambda_{max}$  with  $T_0 = 0$ .

Here the limit exists for all unichain stationary policies from Corollary 4.2.1.1 [2], this limit does not depend on  $x_0$  and satisfies the spectral equation:

$$J_\pi + h_\pi(x) = c(x, \pi(x)) + \sum_{y \in X} p(y|(x, \pi(x))) h_\pi(y). \quad (6)$$

Our aim is to find the optimal policy  $\pi^*$  in  $\Pi$  for which the average cost is smallest. The Bellman optimality equation for the optimal cost per stage is

$$J^* + h^*(x) = \min_q \left( c(x, q) + \sum_{y \in X} p(y|(x, q)) h^*(y) \right).$$

It should be obvious that there exists a stationary policy that reaches any state  $y$  from state  $x$  with positive probability. From there, the existence of an optimal policy whose average cost per stage is  $J^*$  follows from Proposition 4.2.6 in [2].

Finding the solution to this fixed point problem can be done using dynamic programming.

## 4.4 Relation with the Stochastic Game Formulation

One can relate the average cost per stage with the expected cost (related to the expected sojourn time of mobiles) considered in the stochastic game framework.

Indeed by inverting the summation over all rings with the summation over time (which is possible here, using the same trick as in the proof of Little's law) one gets that for a given policy  $\pi$ , the average cost per stage satisfies

$$J_\pi/\Lambda_{max} = C_\pi. \quad (7)$$

The same holds for the optimal policy.

## 5. MEASUREMENT SCENARIOS

We compared the different optimizations, using extensive computations and simulations. This section details the example we numerically solved.

### 5.1 Throughput and model refinements

The different technologies are assumed to use independent frequency bands and hence do not interfere with each other. In the following, we consider an example consisting of 2 cells using different technologies, like WLAN and HSDPA. Both

use TDMA access based multiplexing. The slot allocation may follow different strategies and we follow the bandwidth model given in [6].

We assume that each ring of each technology has a nominal throughput denoted by  $D_c^i$ . This is the bandwidth that a customer located in ring  $i$  would receive if she were alone in the cell.

### 5.1.1 Effective Computations

We consider a simple system made of two cells whose rings superimpose perfectly (hence zones and rings are equal here). One of the cell uses a WLAN protocol while the other one uses the HSDPA technology.

#### Total Throughput with Fair Capacity Share (WLAN).

The WLAN scheduling can be modeled by allocating time slots to mobiles so that they all receive the same throughput, regardless of their location in the cell. Mobiles in remote zones suffer from bad transmission conditions and hence require more time slots for their communication, degrading the throughput of all active mobiles.

The individual bandwidth offered to zone  $i$  in the WLAN cell  $c = 1$  does not depend on  $i$  :

$$d_1^i(\mathbf{n}) = \left( \sum_{k=1}^r \frac{n_1^k}{D_1^k} \right)^{-1}, \quad (8)$$

for a non empty system, the rate being null otherwise.

#### Throughput with equal time shares.

The second cell scheduler (HSDPA),  $c = 2$  is fair in time. This means that all the customers get the medium during the same amount of time. Therefore, at each instant, the capacity is divided by the number of customers currently in the system. The formula for the individual bandwidth on ring  $i$  for cell 2 using a fair time share is

$$d_2^i(\mathbf{n}) = \frac{D_2^k}{\sum_{k=1}^r n_2^k}, \quad (9)$$

for a non empty system and is null otherwise.

### 5.1.2 Dynamic programming

We now want to express the dynamic programming operator from the total average cost equation. The difficulty comes from the reliance of the transition times on the states. Two choices are possible to tackle this. Either include transition state dependence in the operator expression or using uniformization framework. Uniformization framework is most suited for formulation issues but could present some efficiency problems (see [3]), while the other framework is well suited for fast computations but is not very tractable to qualitative studies. This is why we choose the last one in our computational framework. Using this approach, the problem can be solved with any usual methods of dynamic programming such as *value iteration*.

## 5.2 Performance Measures

### 5.2.1 Optimal Policy

The optimal policies are obtained by implementing the value iteration method presented before. We want to ob-

serve how the optimal policies behave according to the input parameters like the rate of arrival, or the cost induced by a rejection. An example is detailed in the next section (Section 6.1).

### 5.2.2 Simulation Results

In order to record a set of characteristics of the system, we have implemented a discrete event simulator. This allows the study of bigger systems than if we had used the exact computation of the stationary distribution of the Markov chain. For instance, the size of a system with 2 rings and 2 cells that could serve 10 mobiles is in the order of  $3 \cdot 10^4$  (because the state depends on the kind of event: either arrival in ring 1 or 2, or departure). Then the size of the transition matrix is the square of this number. By using simulations, we can approximate in little time the stationary distribution of the Markov chain of a system of size  $3 \cdot 15^4$ .

The simulations are done by a program written in C that is run on a laptop with a Intel CoreDuo 1,83 MHz and 1gB of RAM. The question is to know at when to stop a simulation. For this purpose, we first run many simulations and measure the frequency of each state. These frequencies tend to the stationary distribution as time goes on. When the simulations give quite similar frequencies (we fixed a threshold), we stop them and take the current time as stopping time for next simulations. Here, the computing time of a simulation is in the order of 10 seconds, whereas the computation of the optimal policy using value iteration is in the order of 10 minutes.

Finally, for a given policy, we measure  $C_\pi$ , the rejection rate (that is the proportion of arrival mobiles that are rejected by the policy), and the expected total sojourn time of mobiles in the system. The obtained results of these simulations are presented in the next section.

## 6. NUMERICAL RESULTS

In this section, we consider an arrival process of variable rate  $\lambda$ . We assume that the average rate of messages size is  $\frac{1}{\mu} = 3MB$ . The nominal throughput for the WLAN technology is  $D_{WLAN}^1 = 2.6MB/s$  and  $D_{WLAN}^2 = 0.9MB/s$ . In HSDPA networks, the SINR in each ring  $i$  can be approximated by:

$$SINR_i = \frac{1 - \psi}{\alpha\psi + f_i},$$

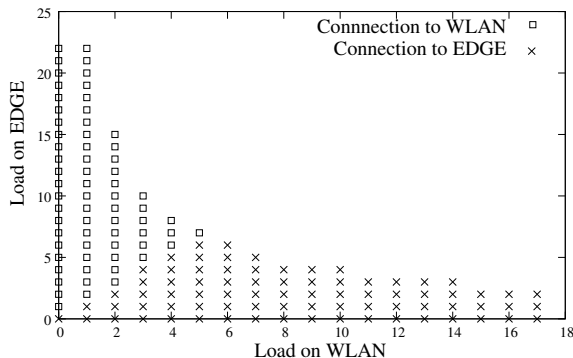
where  $\psi$  is the fraction of power dedicated to common channels,  $\alpha$  is the orthogonality factor and  $f_i$  is the interference factor (the ratio between the total power received by all other base stations and the total received power from the own station). Let us assume the following values for the parameters  $\psi = 0.2$ ,  $\alpha = 0.7$ ,  $f_1 = 0.1$ ,  $f_2 = 1.2$  for two rings. If we assume that the system is able to provide the Shannon capacity:

$$D_{HSDPA}^i = W \log_2(1 + SINR_i),$$

where  $W = 3.84MHz$  is the signal bandwidth. The numerical application provides:  $D_{HSDPA}^1 = 3.1MB/s$  and  $D_{HSDPA}^2 = 0.8MB/s$ . The minimal throughput required by a mobile is  $d_{\min} = 0.25MB/s$ . We assume that the probability for a mobile to arrive in zone 1 is 0.5.

## 6.1 Shape of the Optimal Policy

Figure 2 represents the policy obtained by using the SMDP method. For representation purpose we consider 2 cells and only 1 zone. The nominal throughputs here are  $D_{WLAN} = 2.6MB/s$  and  $D_{HSDPA} = 3.1MB/s$ . The minimal throughput for a connected mobile is  $0.15MB/s$ . The arrival rate is  $\lambda = 8$  and the rejection cost is  $K = 20$ .



**Figure 2: Policy *clair-coop* (obtained with SMDP).**

As there is only one zone, a state is a couple: the first component is the load on the WLAN cell, and the second one is the load on the HSDPA cell. If there were 2 zones, a state would be of dimension 5 (we need to add the fifth component that is the zone in which mobiles arrive). Then, at each state, the policy states whether to accept mobile in WLAN, or HSDPA, or reject it.

In the case of a sole zone, the global sojourn time on a cell is constant, and equal to the nominal rate if the system is non-empty. If both cells are non empty and non full, the policy *MnSVH-coop* does not have any preference between accepting mobiles in HSDPA or WLAN. If the WLAN cell is empty, an arriving mobile would be accepted on the HSDPA cell, and vice-versa is the HSDPA is empty (and the WLAN is not). Further, *MnSVH-coop* policy would not reject any mobile until both systems are full.

The *MSVH-coop* is to accept mobile on the HSDPA cell until it is full and then accept mobiles to the WLAN cell. Finally, the *MnSVH-self* is to accept mobile on the cell which has the lower value of  $D^i/n^i$  until both cells are empty.

The *clair-coop* policy does not correspond to any of these policies. Furthermore, look at state (3,4). If a mobile arrives, in WLAN cell, it would have a sojourn time of  $\frac{4}{2.6}\mu = 0.51s$ , and in HSDPA cell  $0.54s$ . Then, the *MnSVH-self* policy would impose that the mobile connects to the WLAN, while the clairvoyant policy connects it to HSDPA cell.

Finally, we see that the policy rejects the mobiles long before the cells are full (maximum 17 mobiles for WLAN, and 22 for HSDPA). The maximal number of mobiles in both cells is 23 which corresponds to the state (1, 22). This contrasts with any myopic policy, that would accept the mobiles until the system is full.

## 6.2 Influence of the rejection cost (Figures 3 and 4)

Roughly speaking, we observe that for all policies, the global cost  $C_\pi$  for all policies (see, Fig 3(b) and 4(c)) increases with the rejection cost. This is to be expected as the

instantaneous cost is (linearly) increasing with the rejection cost.

More precisely, the cost  $C_\pi$  has two components: the cost of rejection and the cost due to the sojourn time. Figures 3(a), 4(a) and 4(b), illustrate the evolution of the first one. Agreeing to intuition, the rejection rate is decreasing with the rejection cost for all policies. Also, for given values of  $K$  and  $\lambda$ , the rejection rate is lower for zone 1 than zone 2 as this zone benefits of better quality of service (lower expected sojourn time). Now looking more precisely at the different policies, note that in the myopic policies, rejections are due to 2 different causes: either (i) the system is full (no mobile can join while having a higher service rate that  $d_{\min}$  or (ii) the admission controller rejects the call because it would increase the total cost of more than  $K$ . As the admission controller takes its decision only based on the current state of the system (i.e. the load on each zone of each cell) and as the sum of costs is increasing with the load, there is, for each state, a threshold over which any new call from a given zone is accepted (this threshold is much lower in zone 1 than 2). Hence, the total rejection rate due to (ii) (taken as the weighted average over all states) is, for the myopic case, decreasing until it becomes null for a certain  $K$ . Let us call it  $K_{\text{block}}^\pi$ . For such values of  $K$  only remain the rejection due to (i). Therefore, myopic policies have a constant rejection rate above  $K_{\text{block}}$ . Interestingly, from the figures, we note that the value of  $K_{\text{block}}$  is roughly the same for all myopic policies. This is all the more surprising that, as *MSVH-coop* can redistribute the mobiles, at each time epoch, among the cells, then for a given zone load, the global performance that can be achieve is better. Then, its associated  $K_{\text{block}}$  should be higher than those of *MnSVH-coop* and *MnSVH-self*. Yet, from the figures, the improvement is not significant.

Interestingly, *MnSVH-coop* and *MnSVH-self* have very similar rejection rates. Indeed, as the systems have similar constraints, the states in which the system is full are identical. Yet, as the policies are different, the probabilities of being in such states differs, resulting in different rejection rates. Yet, an interesting phenomenon was observed: the rejection rate of *MnSVH-self* is (very) slightly lower than that of *MnSVH-coop*.

Now, *MSVH-coop* and *clair-coop* have both lower rejection rates than the the myopic policies without handover. This is due to two different factors. In *MSVH-coop*, as the mobiles can be redistributed among the cells, the total strategy space is larger and the chances that no association can insures every mobile its  $d_{\min}$  is lower. Hence, for  $K > K_{\text{block}}$  the rejection rate is lower for the policies without handover. Meanwhile, as *clair-coop* anticipates the call arrivals, few calls are rejected because the system is full. Hence, the rejection rate is continuously decreasing with  $K$  without ever reaching its minimum value.

When the rejecting cost is constant, by little Law, the expected sojourn time of mobiles is constant, as confirmed with Figures 3(c) and 4(d). Then, the total cost of the policy for myopic policies is linearly increasing with  $K$  for  $K > K_{\text{block}}$  (Figures 3(b) and 4(c)). Note that, from the figures, under  $K_{\text{block}}$ , all myopic policies have the same performance. For larger values of  $K$ , as the rejecting rate of the *MSVH-coop* is lower than the other two, its slope is smaller and hence, it achieves better performance. Finally, as the rejecting rate of *MnSVH-self* is slightly lower than that of *MnSVH-coop*, then the slope of its  $C$  is smaller as

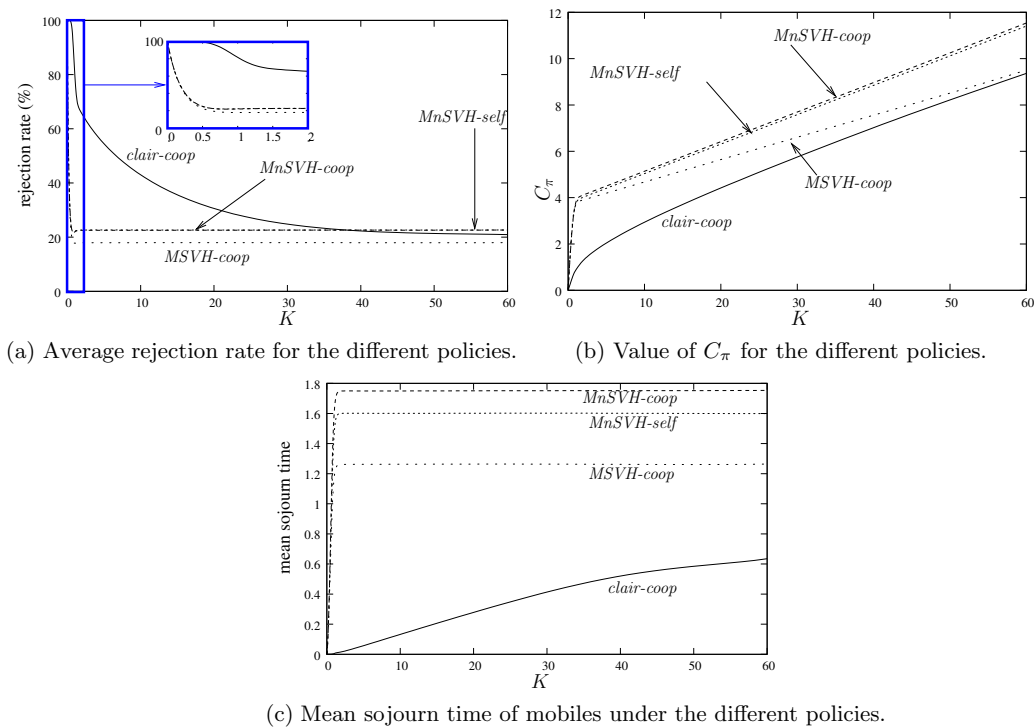


Figure 3: Influence of the rejection cost  $K$  for intermediate arrival rate ( $\lambda = 8$ )

well, and, surprisingly, the selfish approach achieve better performance than the cooperative one. This may be due to the fact that a mobile joining the system would connect to the cell minimizing its own sojourn time, at the expense of the other mobiles present in the system. Then, such mobile would quickly leave the system and free its spot for the next arrival. Further investigation on this behavior is left for future work.

Finally, the rejection rate of the clairvoyant policy is strictly decreasing with the cost  $K$ . Therefore, the mean sojourn time is increasing. In addition, as the system is rarely full, the rejected mobiles are chosen by the policy according to their impact on the system. Hence, the global sojourn time is much lower than for the myopic policies and the resulting cost  $C$  is also lower. The clairvoyant policy benefits are more important for lower values of  $K$  where more flexibility is allowed in the choice of the mobiles to be rejected. This is an interesting result, as it demonstrates that in such cases (relatively small rejection cost values) the benefit gained from forecasting the future of the *clair-coop* policy significantly overrides the technological enhancement of the soft handover that the *MSVH-coop* policy can enjoy.

### 6.3 Influence of the arrival rate (Figure 5)

We observe that for all policies, and for a given rejection cost  $K$ , the global policy cost  $C$  increases with the arrival rate  $\lambda$  (see Figure 5(b)) which agrees with intuition. Indeed, as the arrival rate increases, the systems has greater probability of being full and hence the rejection rate increases for all policies, inducing an higher cost.

Note that the decisions taken by all myopic policies are independent of  $\lambda$ , while the clairvoyant policy smoothly adapts

with the rate. Hence, as the rate increases, the benefits of the clairvoyant policy over the other ones becomes more obvious. Again, we observe that the clairvoyant policy outperforms all myopic policies, including the *MSVH-coop* in spite of its enhanced handover capabilities.

Concerning the myopic policies, Figure 5(a) once again shows that the rejection rates of *MnSVH-self* and *MnSVH-coop* are very similar. Once again a closer look at the simulation results shows that the values are slightly larger for the selfish policy, inducing a slightly smaller cost, although the differences are negligible.

The *MSVH-coop* policy has significantly shorter rejection rate that the other myopic policies, thanks to the larger strategy space. This naturally results in a significant smaller value of its cost (Figure 5(b)).

Finally, interestingly enough, the rejection rate of the clairvoyant policy is comparable to the one of the myopic policies. Yet, because of its wiser choice of the mobiles to be rejected according to the arrival rate, we can note that for an identical rejection rate, the corresponding sojourn time is much lower. Indeed, for instance, for  $\lambda = 10$ , the rejection rate is about 22% for both the *clair-coop* and *MnSVH-self*. Yet, their respective mean sojourn time are respectively 0.4s and 1.6s. Additionally, the clairvoyant policy manages to maintain a constant sojourn time. Hence, as the system is more stressed, the clairvoyant policy smoothly adapts the choice of the mobiles to be rejected so that the quality-of-service is preserved for the mobiles that are granted access.

## 7. CONCLUSION

In this paper, we compared the performance of two fam-

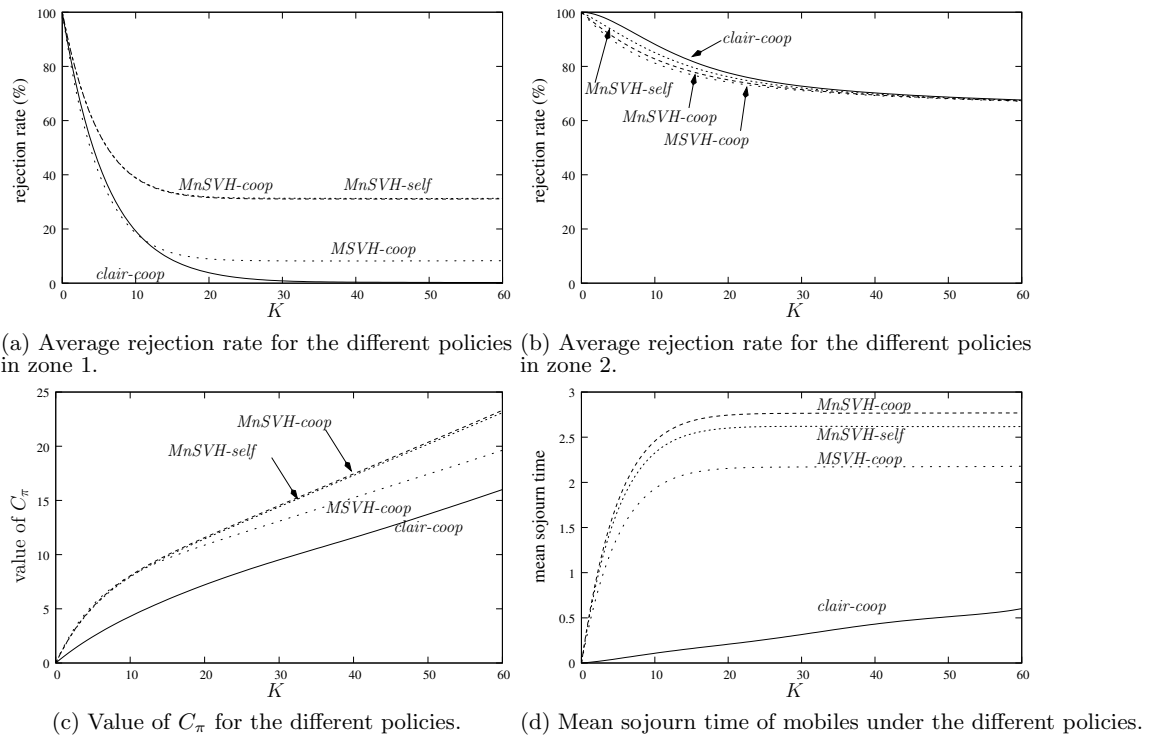


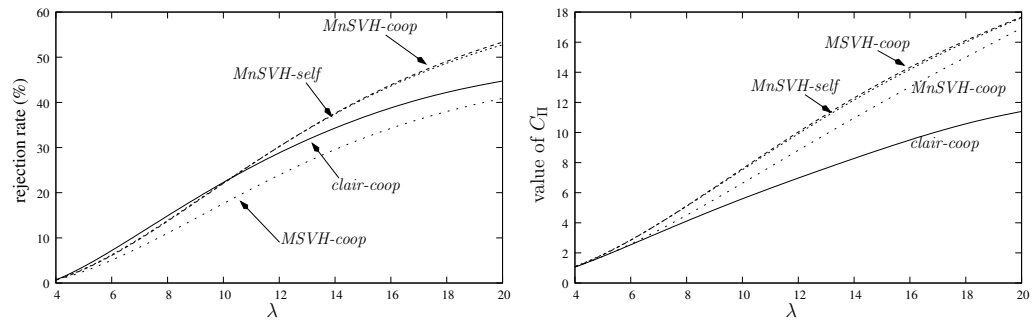
Figure 4: Influence of the rejection cost  $K$  for large arrival rate ( $\lambda = 16$ )

ities of policies in an admission control and optimal association in an heterogeneous wireless network scenario. We observed through simulations that systems with clairvoyant admission control and no soft vertical handover outperforms myopic ones, even when the myopic admission controller is used on a system allowing soft-vertical handover.

We also observed that even though on a static scenario a myopic social optimization has obviously lower (or identical) cost than a selfish optimization, the converse can be observed for their *mean* performance. This demonstrates that when efforts are deployed to improve the performance of algorithms based on a static analysis, nothing can be said about their global performance on a long term view and that care should be taken when extrapolating static case analysis. Our future plan is to better understand the mechanisms that cause the social optimum to be suboptimal in the long run.

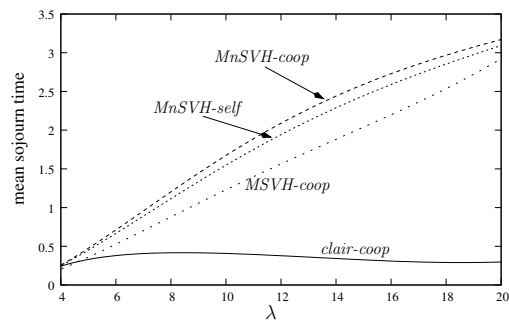
## 8. REFERENCES

- [1] R. Bertin, A. Legrand, and C. Touati. Toward a fully decentralized algorithm for multiple bag-of-tasks application scheduling on grids. In *IEEE/ACM International Conference on Grid Computing (Grid)*, Tsukuba, Japan, 2008.
- [2] D. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific, second edition, 2001.
- [3] F. Beutler and K. Ross. Uniformization for Semi-Markov Decision Processes under stationary policies. *Journal of Applied Probability*, 24(3):644–656, 1987.
- [4] A. Brandt and M. Brandt. Waiting times for M/M systems under state-dependent processor sharing. *Queueing Systems*, 59:297–319, 2008.
- [5] P. Coucheny, C. Touati, and B. Gaujal. Fair and efficient user-network association algorithm for multi-technology wireless networks. In *Infocom*, 2009.
- [6] M. Coupechoux, J.-M. Kelif, and P. Godlewski. Network controlled joint radio resource management for heterogeneous networks. In *IEEE Vehicular Technology Conference (VTC)*, pages 1771–1775, May 2008.
- [7] J. Filar and K. Vrieze. *Competitive Markov Decision Processes*. Springer-Verlag, 1997.
- [8] D. Gross and C. Harris. *Fundamentals of Queueing theory*. Wiley, 2nd edition, 1985.
- [9] M. Kim and H. Park. On achieving short-term qos and long-term fairness in high speed networks. *Journal of High Speed Networks*, 13(3):233–248, 2004.
- [10] D. Kumar, E. Altman, and J.-M. Kelif. User-network association in a WLAN-UMTS hybrid cell: Global & individual optimality. *Research Report, INRIA, Sophia Antipolis, France*, 2006.
- [11] M. Puterman. *Markov Decision Processes Discrete Stochastic Dynamic Programming*. Wiley, 2005.
- [12] L. Shapley. Stochastic games. *Proc. Nat. Acad. Sciences*, 39:1095–1100, 1953.



(a) Average rejection rate for the different policies.

(b) Value of  $C_\pi$  for the different policies.



(c) Mean sojourn time of mobiles under the different policies.

Figure 5: Influence of the arrival rate  $\lambda$  for constant value of  $K$  ( $K = 30$ )