Structured Admission Control Policy in Heterogeneous Wireless Networks with Mesh Underlay

Amin Farbod and Ben Liang
Department of Electrical and Computer Engineering
University of Toronto, Toronto, Ontario, Canada
Email: {afarbod,liang}@comm.utoronto.ca

***Abstract*—In this paper, we investigate into optimal admission control policies for Heterogeneous Wireless Networks (HWN), considering an integration of wireless mesh networks with an overlaying cellular infrastructure. In order to characterize the overflow traffic from the underlaying mesh to the overlay, a Partially-Observable Markov-Modulated Poisson Process (PO-MMPP) traffic model is developed. This model captures the burstiness of the overflow traffic under the imperfect observability of the mesh network states. Then, by modeling the overlay network as a *controlled* PO-MMPP/M/C/C queueing system and obtaining structured decision theoretic results, it is shown that the optimal control policies for this class of HWNs can be characterized as monotonic *threshold curves*. Further, these results are used to design a computationally efficient algorithm to determine the optimal policy in terms of thresholds. Numerical observations suggest that the proposed algorithm is efficient in terms of time-complexity and can drastically reduce the cost of dropped and blocked calls.**

I. Introduction

Heterogeneous wireless networking is an emerging paradigm to simultaneously provide universal coverage and high bandwidth access where available [1]. Heterogeneous Wireless Networks (HWN) consist of several layers of different overlapping wireless access technologies such as an integration of WiMAX or 3G at the overlay with IEEE 802.11 at the underlay [2]. This multi-layer architecture provides users with the option of choosing between different available wireless access technologies based on users' traffic profile, mobility pattern, and QoS preferences.

A major technical requirement for HWNs before they can be fully incorporated into the future wireless communications infrastructure is to support QoS for multimedia services. However, QoS provisioning in these networks is challenging due to the diversity in wireless networking technologies and the existence of mobile users with different communication preferences. Toward this goal, call admission control (CAC) schemes have been used extensively in wireless networks to achieve a desired QoS level. They are in charge of deciding whether to accept or reject call requests and how to reserve resources in a resource-sharing systems.

CAC schemes for *homogeneous* cellular networks have been extensively studied. These schemes can be classified into near-optimal heuristics [3] and decision-theoretic optimal methods [4]. Dynamic Programming (DP) and Markov Decision Processes (MDP) [5] are often used in the design of optimal CAC algorithms. However, for almost all realistic modelings of networking systems, the computational load of finding an optimal policy by MDP algorithms is very high. In fact, there is no known strongly-polynomial time algorithm to solve MDP problems [6]. This can hinder the application of optimal CAC schemes in practical scenarios. Furthermore, in HWNs with their increased complexity, the "curse of dimensionality" problem [7] makes it impractical to directly apply admission control methods that are previously used in homogeneous wireless networks.

A more effective use of DP-based methods is to obtain structural results for optimal control problems [7]–[9]. In this approach, a DP formulation is used to characterize the structure of possible optimal policies. Then, knowledge of the policy structure can be exploited to design efficient numerical methods to find the optimal policy [10]. As an example, in [4], it is shown that the optimal CAC policy for a single cellular Base-Station (BS) is the well-known guard-channel policy, which is fully determined by a single threshold that can be computed by an efficient algorithm. In [11], the authors study a simple two-tier HWN with a single underlay network and show that the optimal CAC policy has a two-dimensional threshold structure. However, such results are not directly applicable to HWNs with a more complex underlay architecture.

In this paper, we consider the characterization of structured optimal CAC policies for HWNs formed by the integration of mesh and cellular networks. An example of such networks is shown in Figure 1. Such HWNs architectures are appealing because mesh networks are expected to be extensively deployed in urban and suburban areas to provide mobile and roaming users with wireless networking services. However, the multiple Access Points (AP) in the mesh underlay network adds one more dimension to the CAC problem space, and furthermore the user movement between different APs creates additional complications in the problem formulation. To the best of our knowledge, there is no known solution to the optimal admission control problem for this architecture.

By making some practical assumptions regarding the service model provided by this class of HWNs, we develop an scalable approximate model to achieve near-optimal performance in the control of these networks. The core assumption is that the priority in modeling of different layers of a HWN should be given to the provisioning of *overflows*. A new or a hand-over call overflows to the overlay when it cannot be accommodated by the target mesh Access Point (AP) at the underlay. It is well known that this overflow traffic is not Poisson [12]. In order to characterize the overflow traffic from the underlaying mesh to the overlay, a Partially-Observable Markov-Modulated Poisson Process (PO-MMPP) traffic model is developed. This model captures the burstiness of the overflow traffic under the the imperfect observability of the mesh network status.

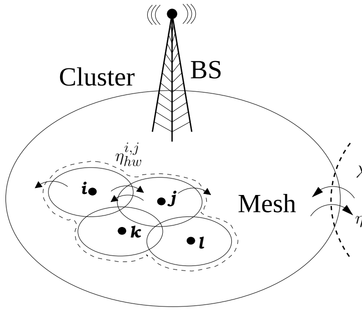

Fig. 1. Cluster traffic arrivals and departures.

Afterwards, the overlay network is modeled as a *controlled* PO-MMPP/M/C/C queuing system, and structural optimal control results are given. The significance of these results is in showing that the optimal control policies for this class of HWNs can be characterized as monotonic *threshold curves*.

Further, the monotonic threshold curve structure of the optimal CAC policy is used to design a computationally efficient Structured Coordinate Search Algorithm (SCSA) to determine the optimal policy in terms of thresholds. Based on this algorithm, we propose a modular CAC scheme, which can be employed in cellular base stations and in mesh-controller nodes to improve the system performance. Through numerical analysis and simulations, we show that the proposed algorithm has a convergence speed much faster than that of the Value Iteration (VI) [13] algorithm and leads to a CAC scheme significantly more cost efficient than the Complete Sharing and Guard-Channel policies.

The rest of the paper is organized as follows. Related works and their contrasts to our contributions in this paper are discussed in Section II. In Section III, the system model and assumptions are discussed. The general framework used in the design of decision-theoretic optimal CAC schemes is explained in Section IV. The development of an approximate overflow model for this HWN architecture is given in Section V. In Section VI, structural results for the control of overlay network are presented. The algorithm to find the structured optimal control policy is given in Section VII. Simulation and numerical results are presented in Section VIII, followed by concluding remarks in Section IX.

II. Related Work and Research Contribution

There are several studies on QoS for HWNs where the conventional guard-channel policy is used and the available channels are divided between handover and new calls coming from different layers. In [14], admission control for voice and data services in a HWNs is considered and an iterative algorithm is proposed to find the dropping and blocking probabilities. In [15], the minimization of linear cost functions is used to find the number of guard channels. In both sets of results the reservation scheme is chosen a priori. This is in contrast to our work, in which we try to rigorously characterize the structure of an optimal reservation policy.

Another related research direction is the modeling of overflow traffic for hierarchical macro/micro-cellular networks. Different models with various complexities are studied in [16]–[19]. These models are built upon prior research on the the characterization of overflow processes for loss queuing systems [20]–[22]. Initially, the overflow traffic has been modeled as a Poisson process [23], however, it was later shown that this approximation is inaccurate, and that by failing to model the burstiness of the overflow traffic, it underestimates the blocking of overflow calls at the overlay [12]. Interrupted Poisson Processes (IPP) and Markov-Modulated Poisson Processes (MMPP) have been used to model the behavior of overflow processes in the context of hierarchical cellular networks [16]–[18], [24].

A common assumption among these studies is that the traffic statistics of all micro-cells are equal, i.e., they have identical arrival and hand-off rates. This reduces analysis complexity by allowing one to focus on one micro-cell, while assuming similar results for the rest. In contrast, for the model considered in this work, because of the asymmetric structure of mesh networks, it is impractical to assume that traffic arrival and handoff rates are similar at different mesh APs. For example, consider the mesh network depicted in Fig. 1. Mesh AP `k` receives handover arrivals from three neighbors, APs `i,j` and `l`. While Mesh AP `i` only receives handovers from two neighbors, APs `j` and `k`. We do not require the assumption of having identical traffic statistics at every mesh AP. Potentially, this can result in intractable complexity. In this work, we propose a method to analyze and control this non-uniform system at a reduced complexity level.

III. System Model

A. Network Model and Underlying Processes

In this work, we focus on a two-tier HWN architecture consisting of an overlay and an underlay, where the underlay can be a mesh network with multiple wireless APs. This basic two-tier entity will be called a *cluster*. An example cluster is shown in Fig. 1. New service requests (more specifically calls in this work) arrive according to a memoryless Poisson process. Call holding times are memoryless with averages of μ_c and μ_w for calls at overlay and underlay. We also assume a memoryless mobility pattern where calls move to neighbor clusters or different layers at exponentially distributed times with rates as denoted in Table I. Here, $\eta_{hw}^{i,j}$ denotes the rate of handover for a call from AP i to AP j as depicted in Fig. 1. The following relations hold between parameters defined in Table I: $\eta_{hw}^{i,j} = \phi(i,j)\eta_h^i$ and $\eta_{hwc}^i = \phi(i,0)\eta_h^i$, where index 0 is reserved for cellular overlay. It is clear that the above memoryless assumptions result in exponential channel holding times [25].

In this paper, we focus on call-level QoS, which is common in CAC literature. Further, the fixed channel allocation (FCA) scheme is used. FCA easily applies to various wireless technologies with channel being frequency, time-slot or code assignment. Let C_c and C_w denote the capacity of overlay BS and underlay Mesh APs measured in basic bandwidth units (BBU). We assume that all multimedia calls require a single BBU. This corresponds to the single service systems studied in [7]. In this work, we only consider multimedia call classes which require a certain amount of bandwidth in a guaranteed-service fashion. For data services, if they require guaranteed bandwidth, they can be modeled along with the multimedia

TABLE I
SYMBOLS USED AND THEIR DESCRIPTION.

Symbol	Description
C_c	Capacity of overlay cellular BS (in BBU)
C_w	Capacity of underlay mesh AP (in BBU)
μ_c	Call holding time rate at overlay
μ_w	Call holding time rate at underlay
λ_{nc}	Arrival rate of new calls to overlay
λ_{hcc}^{in}	Handoff arrival rate from neighboring overlay cells
λ_{nw}^{i}	Arrival rate of new calls to underlay AP *i*
λ_{ho}^{i}	Rate of handoff overflows from AP *i*
λ_{no}^{i}	Rate of new call overflows from AP *i*
λ_{hw}^{i}	Rate of calls leaving AP *i*
η_{hcc}	Handoff rate to neighboring overlay cells
η_{h}^{i}	Handoff rate out of underlay AP *i*
$P_{D}^{(i)}$	Handoff calls dropping probability at AP *i*
$P_{B}^{(i)}$	New calls blocking probability at AP *i*
$\phi(i,j)$	Fraction of calls leaving AP *i*, going to AP *j*

calls, and if they require best-effort services, they only utilize the available unused bandwidth at any time instant, and as such do not affect our analysis of QoS for guaranteed-service call requests.

B. Call Handling Policy

In [11], we assumed tight coupling between different layers of wireless network [26] in a cluster, where the management of different layers is centralized. However, when overlay can comprise mesh networks with multiple APs, it is impractical to assume centralized control, because the network management traffic overhead will be excessively high. In what follows, we assume that there exists a control unit that makes the CAC decision for calls coming to the overlay based on the state of the overlay and some partial knowledge about the status of the underlay mesh network.

We assume that once a call overflows to overlay, it stays at that layer until its completion, i.e., repacking is not allowed. This common assumption allows one to analyze the underlay independently of the overlay network [17], [18]. Note that the no-repacking policy serves as an indirect scheme for classification and layer assignment of users with different velocities. In particular, calls made by a highly mobile user is more likely to request an overflow and hence more likely to be eventually assigned to the overlay during the call period, which clearly is a desirable outcome for system efficiency.

When a new call request arrives to the system, if it is within the double coverage area (DBCA), it will seek admission to the underlay network first. If it can not be accommodated by the underlay, it overflows to the overlay network. Also, when a handover call moves from an underlay AP to either the single coverage area (SCA), or another underlay AP where it can not be accommodated, it overflows to the overlay network. For a new call, if it is initiated within the SCA, it will directly seek admission to the overlay.

C. Underlay Mesh Network Operation

In this paper, we assume that the capacity offered by every mesh node to its local users is fixed. For the modular CAC scheme proposed in this paper, it is assumed that every mesh AP makes the decision of CAC independently. As no repacking of overlay calls are allowed, mesh APs receive no traffic load from the overlay, and it suffices for them to make the CAC decision based on the traffic statistics of their own and their immediate neighbors in the mesh topology. For this purpose, any of the previously studied CAC schemes can be used [3], [4].

TABLE II
REJECTION COSTS FOR NEW AND HANDOFF CALLS .

Rate	Rejection Cost	Description
λ_{nc}	C_{BNC}	Blocking of new calls
λ_{no}^{i}	C_{BNO}	Blocking of new call overflows
λ_{hcc}^{in}	C_{DCC}	Dropping of cell-to-cell handovers
λ_{ho}^{i}	C_{DHO}	Dropping of handover overflows

IV. DYNAMIC-PROGRAMMING FRAMEWORK

MDP methods are used to find the optimal policy to control a markovian random process over time to achieve a certain optimization goal [13]. An MDP is determined by four components: state space S, action space A, state transition probabilities $P(.)$ (or transition rates $Q(.)$ in case of continues time processes), and a cost function $C(.)$. The solution to an MDP is called a policy or rule. A policy maps the state space to actions $\Psi : S \rightarrow A$, such that the optimization goal is achieved. A large class of policies, in which the decision is independent of time, are called stationary policies.

In our event-based DP, we associate costs to undesirable control decision events. These costs correspond to the dropping or blocking of incoming calls. A list of rejection costs for different call types is given in Table II. Note that the cost of dropping an AP-to-AP handover is not considered, since a rejected AP-to-AP handover call simply creates an overflow handover to the overlay. Throughout the rest of this paper, every call type is called a *class*.

In the study of CAC schemes, the most common performance criterion is the minimization of a total cost (objective) function which is referred to as MINOBJ in [4]. We formally define MINOBJ as

$$\text{MINOBJ}: \quad \min_{\Psi} g_{\Psi} = \sum_{l=1}^{L} C_R^{(l)} \lambda_l P_R^{(l)} \tag{1}$$

where $C_R^{(l)}$ is the cost of rejecting a call request of class l, λ_l is the arrival rate of class l calls, $P_R^{(l)}$ is the rejection (either blocking or dropping) probability for that class and L is the total number of call classes. Our objective is to minimize the average cost per unit of time for an infinite-horizon non-discounted problem. This reflects our concern about long-run QoS performance. In addition, since the decision epochs can be at any randomly distributed time, a Semi-Markov Decision Process (SMDP) model is used [27].

In what follows, we present a general framework that will be used throughout the rest of the paper. We start with a finite-horizon optimal cost function, and we show that the solution to the infinite-horizon problem has the same structure. Let us denote by $V_k(s)$ the minimum expected cost function for a k-stage problem with the initial state s. Also, we use E_A to denote the set of arrival events which are controllable and can potentially be rejected at the costs defined in Table II, and E_N to denote the set of internal transitions and departures for which no action needs to be taken. Using the uniformization technique [27], we can write $V_{k+1}(s)$ recursively as

$$V_{k+1}(s) = \frac{1}{v_{max}}\Big\{ \sum_{e\in E_A} q_e\big[\min\{\Delta V_k(\psi_e s), C_{B(e)}\} + V_k(s)\big] + \sum_{e\in E_N} q_e V_k(\psi_e s) + (v_{max} - v_{out}(s))V_k(s)\Big\} \quad (2)$$

where $v_{out}(s)$ is the rate of going out of state s, and v_{max} is the uniformization parameter such that $v_{max} \geq v_{out}(s)$ for every s. Here, operator ψ_e acts on state s and returns the resultant state if event e was to be admitted. The Δ symbol for an operator ψ is defined as

$$\Delta V_K(\psi_e s) = V_K(\psi_e s) - V_K(s). \quad (3)$$

Also, q_e is the rate of transition for events of type e. Equation (2) consists of three terms, each reflecting one possible event. The first term reflects arrivals to the cluster and the second term accounts for departures and internal non-controllable transitions. The last term is due to the uniformization technique where staying in the same state is possible. Note that overflow events despite being internal are controllable, and as such belong to set E_A. In the first term, $\Delta V_K(\psi_e s)$ is the cost of admitting a call whose arrival is triggered by event e. If this cost is less than blocking cost $C_{B(e)}$, the call will be admitted and otherwise it gets rejected. More general information on similar frameworks can be found in [4], [28].

It has been shown in [27] that for average-cost problems with finite S and A, the optimal policy is stationary. Further, we are only interested in stationary policies which result in irreducible chains. According to Theorem (6.6.2) in [27], for irreducible and aperiodic Markov decision processes the difference of upper and lower bounds of $V_{k+1}(s) - V_k(s)$ for different values of s converges to the optimal average cost per unit time when $k \to \infty$. More formally, given

$$M_k = \max_{t\in S} \ \{V_{k+1}(t) - V_k(t)\}, \quad (4)$$

the optimal cost g_Ψ as defined in (1) is equal to $g_\Psi = \lim_{k\to\infty} M_k$. Furthermore, Theorem (6.6.1) in [27] implies that the structural results for $V_k(s)$ as defined in (2) will hold for the optimal per-unit-time average cost function if the underlying Markov decision process is irreducible and aperiodic. This means the structure of the optimal policy to achieve MINOBJ is the same as the policy which minimizes $V_k(s)$.

V. Approximate Model for Underlay Overflow

If we were to model the system by a global state, the cardinality of state space S will be $C_c(C_w)^M$, which is excessively high. To reduce the complexity of the control policy, we propose an approximate model which focuses on the overflow traffic from the mesh underlay.

A. Analysis of Mesh Underlay

A conventional simplifying assumption in the analysis of micro-cellular networks is to equate outgoing and incoming handoff rates for every micro-cell [17]. In our model, due to the asymmetric nature of the mesh network, we can not use such simplifications.

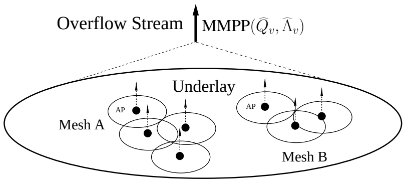

Fig. 2. Underlay overflow model.

Let λ_T^i denote the total traffic arrival to underlay AP i; we have

$$\lambda_T^i = \lambda_{nw}^i + \sum_{v=1, v\neq i}^{M} \lambda_{hw}^v \phi(v,i). \quad (5)$$

For AP i, we define the utilization factor as $\rho_i = \lambda_T^i/(\mu_w + \eta_h^i)$ and $\delta_i = \lambda_{nw}^i/\lambda_T^i$. As we have explained in Section III, the CAC scheme at the underlay can be chosen arbitrarily among any of the available schemes such as a guard-channel policy or alike [3], [4]. For a guard-channel CAC policy with g guard channels the blocking performance of AP i can be found as [4]

$$\begin{aligned} P[m_i = k] &= \frac{\rho_i^k \Pi_{j=1}^k \gamma_j}{k! \sum_{l=0}^{C_w} (\rho_i^l \Pi_{j=1}^l \gamma_j / l!)} \\ P_B^{(i)} &= \sum_{k=C_w-g}^{C_w} P[m_i = k] \\ P_D^{(i)} &= \ P[m_i = C_w], \end{aligned} \quad (6)$$

where $\gamma_j = 1 - \delta_j \ I_{\{j \geq C_w - g\}}$, and the indicator function I_θ equals 1, only if θ is true. $P[m_i = k]$ is the probability of having k active calls in AP i. If $g = 0$, no control is exercised to give priority to handover calls, and (6) reduces to the well-known Erlang-B loss formula.

The probability that a call in AP i will require a handover before termination is $R_h^{(i)} = \eta_h^i/(\eta_h^i + \mu_w)$. For the total rate of calls leaving AP i due to handover, we have

$$\lambda_{hw}^i = R_h^{(i)} \{(1 - P_D^{(i)}) \sum_{v=1, v\neq i}^{M} \lambda_{hw}^v \phi(v,i) + (1 - P_B^{(i)})\lambda_{nw}^i\}. \quad (7)$$

This forms a set of non-linear equations, with solution being the set of λ_{hw}^i for all APs. Due to possible existence of loops in the network topology, the determination of $P_B^{(i)}, P_D^{(i)}$ and λ_T^i may involve iterations. For example, the Erlang Fixed-Point approximation has been used to find the traffic intensities and blocking probabilities in loss networks [22], [29]. We start with a set of values for $P_B^{(i)}$ and $P_D^{(i)}$, and solve (7) to find the set of λ_{hw}^i. Then, (6) is used to recompute a new set of $P_B^{(i)}$ and $P_D^{(i)}$. This iteration is repeated until desired accuracy is reached. The proof for uniqueness and convergence of this iterative technique to the proper solution is given in [22].

In the rest of this paper, for illustration simplicity, we assume that no CAC scheme (e.g., guard-channel policy) is used in the underlay, which implies that $P_D^{(i)} = P_B^{(i)}$. It is notable that the framework presented below can be easily extended to accommodate a wide set of underlay CAC schemes.

B. Analysis of Overflow Traffic

Three events trigger the overflow of a call from an underlay AP to overlay: 1)when an active call (mobile station) associated with an AP leaves the double-coverage area (DBCA), 2)when a new call arrives to a blocking AP and 3)when a handover call comes to an AP that drops it. The overflow generated by the events of the first type forms a Poisson stream with a total rate of

$$\lambda_{po} = \sum_{v=1}^{M} \lambda_{hw}^{v} \phi(v,0). \tag{8}$$

However, for events of types two and three, the overflow is not uniformly distributed over all AP states, and as studied previously it has a bursty nature [16]–[19]. The Interrupted Poisson Process (IPP) and Markov Modulated Poisson Process (MMPP) traffic models have been used to deal with such streams. As a first approximation, we use an MMPP model. The overflow modeling is depicted in Fig. 2. If AP i is in the blocking states, e.g., $m_i \in \{C_w - g, \ldots, C_w\}$, new calls coming with rate $\lambda_{no}^{i} = \lambda_{nw}^{i}$, will overflow. If the AP is in dropping mode, e.g., $m_i = C_w$, then in addition to the new call's overflow, incoming handovers will also overflow. The rate of handover overflow from AP i when it is in the dropping mode is

$$\lambda_{ho}^{i} = \sum_{v=1, v \neq i}^{M} \lambda_{hw}^{v} \phi(v,i). \tag{9}$$

An MMPP(Q, Λ) with N states is identified by two parameters, Q an $N \times N$ matrix, and $\Lambda = \mathtt{diag}(\lambda_1, \ldots, \lambda_N)$, a diagonal matrix of size $N \times N$. Q is the infinitesimal generator for the MMPP underlying process, and Λ determines the rate generated at each state. Let MMPP(Q_i, Λ_i) denote the model for the bursty overflow from AP i. Here, traffic overflows with rate $\lambda_{ho}^{i} + \lambda_{no}^{i}$ whenever the system is at state $m_i = C_w$. At other states the overflow rate is zero, i.e., $\Lambda_i = \mathtt{diag}(0, \ldots, 0, \lambda_{ho}^{i} + \lambda_{no}^{i})$.

We adapt the technique of representing every of these MMPPs with an equivalent IPP. An IPP(λ, α, β) has three parameters. It is either in state ON or in state OFF. If it is in state ON, it generates a Poisson stream with rate λ. $1/\alpha$ is the mean of on-time period, and $1/\beta$ is the mean of off-time period. An IPP can be expressed as an MMPP with two states as

$$Q_{IPP} = \begin{bmatrix} -\beta & \beta \\ \alpha & -\alpha \end{bmatrix}, \quad \Lambda_{IPP} = \begin{bmatrix} 0 & 0 \\ 0 & \lambda \end{bmatrix}. \tag{10}$$

To find the equivalent IPP of an MMPP, we employ the technique used in [21]. The method is to match the first two noncentral moments of the arrival rate of the MMPP and an appropriately chosen time constant. Let us denote by π_{Q_i} the stationary probability vector associated with Q_i such that $\pi_{Q_i} Q_i = 0$ and $\pi_{Q_i} \mathbf{e} = 1$. The nth noncentral moment of the instantaneous rate of MMPP(Q_i, Λ_i) is $m^{(n)} = \pi_{Q_i} \Lambda_i^{n} \mathbf{e}$, which gives us the variance as $v = m^{(2)} - (m^{(1)})^2$ and the time constant τ_c as

$$\tau_c = v^{-1} \left[\pi_{Q_i} \Lambda_i (\mathbf{e}\, \pi_{Q_i} - Q_i)^{-1} \Lambda_i \mathbf{e} - (m^{(1)})^2 \right]. \tag{11}$$

Now, the parameters for the equivalent IPP$(\lambda_i, \alpha_i, \beta_i)$ can be determined as

$$\begin{aligned} \lambda_i &= \lambda_{ho}^{i} + \lambda_{no}^{i} \\ \alpha_i &= [1 - \pi_{Q_i}(C_w)]/\tau_c, \quad \beta_i = \pi_{Q_i}(C_w)/\tau_c. \end{aligned} \tag{12}$$

The total overflow from underlay can be modeled as the superposition of these M independent IPPs. If we cast the ith IPP as a two state MMPP as given in (10) with $(\widetilde{Q}_i, \widetilde{\Lambda}_i)$, the aggregate model can be represented by an MMPP(Q_T, Λ_T) with

$$\begin{aligned} Q_T &= \widetilde{Q}_1 \oplus \widetilde{Q}_2 \oplus \cdots \oplus \widetilde{Q}_M \\ \Lambda_T &= \lambda_{po} I + \widetilde{\Lambda}_1 \oplus \widetilde{\Lambda}_2 \oplus \cdots \oplus \widetilde{\Lambda}_M \end{aligned} \tag{13}$$

where operator $\oplus$ is the Kronecker sum [21]. Here, Q_T has the size of $2^M \times 2^M$; two states for every IPP being ON or OFF. Note that if the underlay APs had similar statistics, as assumed in almost all previous studies, this MMPP could be simplified to an MMPP with M+1 states. However, this is not the case here.

The large size of the system state space as shown above would render an exact control algorithm prohibitively complex for practical applications. For example, to fully utilize Q_T, the system would need, at every decision epoch, the exact status of every underlay AP, which clearly is not a scalable approach.

In what follows, we adopt a state-space reduction approach, to consider only the total number of dropping/blocking APs. In practice, this may correspond to an overlay controller that has only a partial view of the underlay mesh network, such that it is made aware of dropping/blocking APs due to overflow new and handover calls but otherwise has no additional information on the state of all APs. Formally, if the state of the ith IPP is denoted by x_i (with $x_i = 1$ when IPP is ON), then the underlay state can be expressed as $\vec{u} = (x_1, x_2, \ldots, x_M)$. We define the *degree* operator on $\vec{u}$ as $\omega(\vec{u}) = \sum_{i=1}^{M} x_i$. At state $\vec{u}$, the overlay controller only knows the total count of ON IPPs, which we denote by random variable $N = \omega(\vec{u})$.

We define $U = \{\vec{u}\,|x_i \in \{0,1\},\ i = 1, \ldots, M\}$ as the state space of the underlay network. U can be partitioned into M+1 disjoint and exhaustive subsets $U(k)$ based on our observation of $N = k$. We have

$$U(k) = \{\vec{u} \in U \mid \omega(\vec{u}) = k\}. \tag{14}$$

Given that the control decision has to be made based on the observation of N, and not $\vec{u}$, we can derive a new PO-MMPP$(\widehat{Q}_v, \widehat{\Lambda}_v)$ model based on MMPP(Q_T, Λ_T). In this new model, states are associated with observations of N, and hence, the resultant MMPP has M+1 states. We note that at every transition of $\vec{u}$, $\omega(\vec{u})$ only changes by one. Therefore, $\{N(t)\}$ is a birth-death Markov chain. In order to characterize $\widehat{Q}_v$ and $\widehat{\Lambda}_v$, let us denote the birth rates by vector $\vec{\alpha}_v = [\alpha_v^{(1)}, \ldots, \alpha_v^{(M)}]$, the death rates by $\vec{\beta}_v = [\beta_v^{(1)}, \ldots, \beta_v^{(M)}]$ and the arrival rates as $\hat{\lambda}_v^{k}$ for $k = 0, \ldots, M$.

Let us denote by $\lambda_{\vec{u}}$ the rate of MMPP(Q_T, Λ_T) at state $\vec{u}$. Let π_t be the stationary probability vector associated with Q_T such that $\pi_t Q_T = 0$ and $\pi_t \mathbf{e} = 1$, and $\pi_{U(k)}$ be the lump sum probability of being in aggregate state $N = k$, given as $\pi_{U(k)} = \sum_{\vec{v} \in U(k)} \pi_t(\vec{v})$. If we observe that $N = k$, the expected instantaneous overflow rate is

$$\hat{\lambda}_v^k = E[\lambda_{\vec{u}}|\vec{u}\in U(k)] = \sum_{\vec{u}\in U(k)} \lambda_{\vec{u}} \frac{\pi_t(\vec{u})}{\pi_{U(k)}}. \tag{15}$$

We also have to determine what fraction of $\hat{\lambda}_v^k$ is new calls and what fraction is formed by handover calls, as these streams have different rejection costs. Let us denote by $\lambda_{ho}(\vec{u})$, the total rate of handover overflows and $\lambda_{no}(\vec{u})$ the total rate of new calls overflow at state $\vec{u}$. We have

$$\lambda_{ho}(\vec{u}) = \lambda_{po} + \sum_{i=1}^{M} x_i\, \lambda_{ho}^i, \quad \lambda_{no}(\vec{u}) = \sum_{i=1}^{M} x_i\, \lambda_{no}^i. \tag{16}$$

The total overflow rates for new and handover calls for when PO-MMPP$(\widehat{Q}_v, \widehat{\Lambda}_v)$ is in state k can now be calculated as

$$\lambda^{(k)}_{(h\backslash n)v} = \sum_{\vec{u}\in U(k)} \hat{\lambda}_v^k \lambda_{(h\backslash n)o}(\vec{u})/(\lambda_{ho}(\vec{u}) + \lambda_{no}(\vec{u})). \tag{17}$$

To find birth-death rates, we use the Markov chain state aggregation method proposed in [30]. The transition rate from subset $U(i)$ to $U(j)$ denoted by $q_v(i,j)$ can be found as

$$q_v(i,j) = \sum_{\vec{x}\in U(i)} \sum_{\vec{y}\in U(j)} q_t(\vec{x},\vec{y}) \frac{\pi_t(\vec{x})}{\pi_{U(i)}} \tag{18}$$

where $Q_T = [q_t(\vec{x},\vec{y})]$. The birth-death rates can be calculated as

$$\alpha_v^{(k)} = q_v(k-1,k), \quad \beta_v^{(k)} = q_v(k,k-1). \tag{19}$$

Equations (15) and (19) completely determine the underlay overflow process as PO-MMPP$(\widehat{Q}_v, \widehat{\Lambda}_v)$.

VI. Structured Control of Overlay BS

In this section, we study structural results for optimal control of the overlay cellular BS and model the overlay network as a controlled PO-MMPP/M/C/C queuing system.

In this system model, controlled transitions at the overlay are 1)new call arrivals to overlay, 2)new call overflows from underlay, 3)handover call arrivals to overlay from neighbor clusters, and 4)handover call overflows from underlay to overlay. With the rejections costs of C_{BNC}, C_{BNO}, C_{DCC} and C_{DHO}, respectively. We refer to each of these request types as a call class. We define the system state as $s = (i,n)$, where i is the number of calls at the overlay and n is the total number of dropping/blocking APs at the underlay. We also define state operators of

$$\begin{array}{l} \mathcal{A}s : s = (i,n) \to s' = (i+1,n) \\ \mathcal{D}s : s = (i,n) \to s' = (i-1,n) \\ \mathcal{Q}s : s = (i,n) \to s' = (i,n+1) \\ \mathcal{R}s : s = (i,n) \to s' = (i,n-1). \end{array} \tag{20}$$

These operators are boundary sensitive, which means that they do not map a state to a point outside the system state space. For example, if $s + (0,1) \notin S$, then $\mathcal{Q}s$ is not a proper operation. Here, the state space S is defined as

$$S = \{s = (i,n) : 0 \le i \le C_c,\ 0 \le n \le M\}. \tag{21}$$

Extending the framework presented in Section IV, we can write the optimal cost function $V_k(s)$ for this system as

$$\begin{aligned} V_{k+1}(s) \ = \ \frac{1}{v_{max}} \{ & \lambda_{hcc}^{in} \min[\Delta V_k(\mathcal{A}s), C_{\text{DCC}}] \\ & + \lambda_{hv}^{(n)} \min[\Delta V_k(\mathcal{A}s), C_{\text{DHO}}] \\ & + \lambda_{nv}^{(n)} \min[\Delta V_k(\mathcal{A}s), C_{\text{DNO}}] \\ & + \lambda_{nc} \min[\Delta V_k(\mathcal{A}s), C_{\text{BNC}}] \\ & + \alpha_v^{(n)} \Delta V_k(\mathcal{Q}s) + \beta_v^{(n)} \Delta V_k(\mathcal{R}s) \\ & + i(\mu_c + \eta_{hcc}) \Delta V_k(\mathcal{D}s) + v_{max} V(s) \} \end{aligned} \tag{22}$$

where $v_{out}(s) =$ is the sum of all transition rates (q_e) out of state s

$$\begin{aligned} v_{out}(s) = \ & \lambda_{nc} + \lambda_{nv}^{(n)} + \lambda_{hv}^{(n)} + \lambda_{hcc}^{in} \\ & + i(\mu_c + \eta_{hcc}) + \alpha_v^{(n)} + \beta_v^{(n)}. \end{aligned} \tag{23}$$

The following Lemmas are required to obtain structural results for $V_k(s)$, where Lemma 1 is needed to prove Lemma 2. The proofs are omitted in this version due to lack of space; interested readers are referred to the authors' supplemental technical report available online.

Lemma 1: $\lambda_{nv}^{(n)}$ and $\lambda_{hv}^{(n)}$ are both monotonically non-decreasing in n.

Lemma 2: $V_k(s)$ has the following properties:

A) $\Delta V_k(\mathcal{A}s) \ge 0$
B) $\Delta V_k(\mathcal{A}^2 s) \ge \Delta V_k(\mathcal{A}s)$
C) $\Delta V_k(\mathcal{A}\mathcal{Q}s) \ge \Delta V_k(\mathcal{A}s)$.

In the above, Property A is needed to prove properties B and C. Intuitively, Property B states that the cost function of the system, and its differences, monotonically increase with the number of calls being accommodated by the system. Property C states that the differences of cost function monotonically increase with the number of dropping/blocking APs at the underlay.

Definition 1: A *threshold policy* is a CAC policy in which resource requests of class $\mathbf{r}$ are admitted if and only if, the system state is not greater than the threshold for class $\mathbf{r}$, i.e., $s_i \le \text{Th}[\mathbf{r}]$.

We denote by $\Pi = \{\text{Th}(L,W)\}$ the class of threshold polices. Here, $L = 4$ is the number of controlled call classes entering overlay, and W is number of possible states the uncontrollable variable can assume. In this scenario, $W = M+1$. A policy $\Psi \in \Pi$ fully determines the CAC algorithm in terms of thresholds.

Theorem 1: The optimal control policy for the system model given in (22) is a threshold policy.

Proof: Let us assume that a call of class l arrives to the system with the rejection cost of $C_R^{(l)}$, when the system state is $s = (i,n)$. If the new call is admitted, the increase in the optimal cost function is $\Delta V_k(\mathcal{A}(i,n))$. We show that the CAC decision can be expressed in terms of thresholds determined by $\Delta V_k(\mathcal{A}(i,n))$ and $C_R^{(l)}$.

If $\exists i_0$ such that $\Delta V_k(\mathcal{A}(i_0,n)) < C_R^{(l)}$ and $\Delta V_k(\mathcal{A}(i_0,n)) \ge C_R^{(l)}$, then Lemma 2.B asserts that for all $i \le i_0$ we have $\Delta V_k(\mathcal{A}(i,n)) < C_R^{(l)}$, and for all $i \ge i_0$ we have $\Delta V_k(\mathcal{A}(i,n)) \ge C_R^{(l)}$. This means that i_0 is the state after which the cost of admitting a call of class l becomes more than rejecting it at the cost of $C_R^{(l)}$. Therefore, $\text{Th}(l,n) = i_0$. If such i_0 does not exist, either for all $0 \le i \le C_c$ we have $\Delta V_k(\mathcal{A}(i,n)) \ge C_R^{(l)}$ and calls of class l are rejected in all states, or for all $0 \le i \le C_c$ we

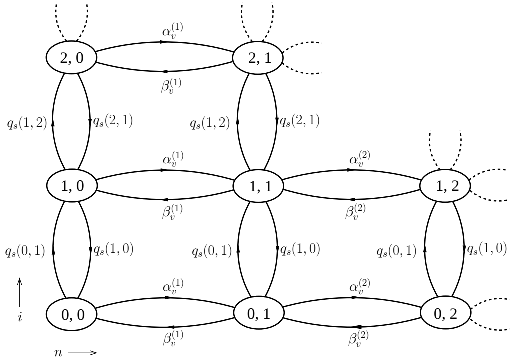

Fig. 3. 2D Markov chain model for overlay analysis.

have $\Delta V_k(\mathcal{A}(i,n)) < C_R^{(l)}$ and calls of class l are admitted in all states. ■

In the next theorem, we claim that for every call class, the threshold is a monotonic curve. It is monotonic in that it decreases with an increase in n.

Theorem 2: The optimal threshold policy for the system model given in (22) belongs to the class of monotonic threshold curves.

Proof: We have to show that if $n_2 \leq n_1$, then $\mathrm{Th}(l,n_1) \leq \mathrm{Th}(l,n_2)$. Defining $i_1 = \mathrm{Th}(l,n_1)$ and $i_2 = \mathrm{Th}(l,n_2)$, we have

$$\Delta V_k(\mathcal{A}(i_1,n_1)) < C_R^{(l)} \leq \Delta V_k(\mathcal{A}(i_1+1,n_1)) \quad (24)$$

$$\Delta V_k(\mathcal{A}(i_2,n_2)) < C_R^{(l)} \leq \Delta V_k(\mathcal{A}(i_2+1,n_2)) \quad (25)$$

We want to show that $i_1 \leq i_2$. For contradiction, assume $i_1 > i_2$; then Lemma 2.B gives us $\Delta V_k(\mathcal{A}(i_1,n_2)) \geq \Delta V_k(\mathcal{A}(i_2,n_2))$, because $(i_1,n_2) = \mathcal{A}^k(i_2,n_2)$ where $k = i_1 - i_2 > 0$. Knowing $i_1 \geq i_2 + 1$ and comparing with (25) and applying Lemma 2.B, we have $\Delta V_k(\mathcal{A}(i_1,n_2)) \geq C_R^{(l)}$. Also, from (24) we have $\Delta V_k(\mathcal{A}(i_1,n_1)) < C_R^{(l)}$. Which together gives us $\Delta V_k(\mathcal{A}(i_1,n_2)) \geq \Delta V_k(\mathcal{A}(i_1,n_1))$. Given Lemma 2.C, this implies that $n_2 \geq n_1$ which contradicts our initial assumption. ■

Definition 2: A *threshold policy* is an ordered threshold policy when we have

$$\mathrm{Th}(l,n) \leq \mathrm{Th}(l',n) \quad \text{If } C_R^{(l)} \leq C_R^{(l')}.$$

Theorem 3: The monotonic threshold curves from Theorem 2 are ordered.

The proof is omitted due to space limitation. In the next section, we use the above structural properties to design an efficient computational algorithm.

VII. Structured Coordinate Search Algorithm

Depending on the system size, the computation cost of finding an optimal CAC policy can be very high. The optimal policy can be computed either by using any of Value Iteration (VI), Policy Iteration (PI), or Linear Programming (LP) techniques, or directly, through explicit formulation of cost criterion and combinatorial search. In this section, we propose an efficient algorithm which determines the parameters of an optimal threshold policy using combinatorial localized search algorithms [31].

Given a CAC policy Ψ, the system can be analyzed as a Markov chain with states (i,n) as depicted in Fig. 3. This Markov chain still has a large state space. However, observing the separation between its two dimensions, we can simplify the computation of the stationary state distribution

$$\pi_s(f,k) = P[i=f|n=k]\ P(n=k). \quad (26)$$

Let us first find $P(n=k)$. From Section V, we know that the PO-MMPP model of the overflow traffic is a birth-death process. We have

$$P(n=k) = \frac{\Pi_{i=1}^{k} \frac{\alpha_v^{(i)}}{\beta_v^{(i)}}}{1 + \sum_{j=1}^{M} \Pi_{i=1}^{j} \frac{\alpha_v^{(i)}}{\beta_v^{(i)}}}. \quad (27)$$

It can be seen from Fig. 3 that conditioned on n the transitions for i form a birth-death process. We have

$$P[i=f|n=k] = \frac{\Pi_{h=1}^{f} \frac{q_{s|n}(h-1,h)}{q_{s|n}(h,h-1)}}{1 + \sum_{j=1}^{C_c} \Pi_{h=1}^{j} \frac{q_{s|n}(h-1,h)}{q_{s|n}(h,h-1)}}. \quad (28)$$

For the system model given in (22), $q_{s|n}$ can be found as

$$\begin{aligned} q_{s|n}(i,i+1) =& \lambda_{nc} I_{\mathrm{th}}(i,n,1) + \lambda_{nv}^{(n)} I_{\mathrm{th}}(i,n,2) \\ &+ \lambda_{hcc}^{in} I_{\mathrm{th}}(i,n,3) + \lambda_{hv}^{(n)} I_{\mathrm{th}}(i,n,4) \\ q_{s|n}(i,i-1) =& i(\mu_c + \eta_{hcc}) \end{aligned} \quad (29)$$

where call classes $l = 1,2,3,4$, are enumerated according to Table II, and $I_{\mathrm{th}}(i,n,l)$ is the admission region indicator function given as

$$I_{\mathrm{th}}(i,n,l) = \begin{cases} 1 & \text{if } i \leq \mathrm{Th}(l,n) \\ 0 & \text{otherwise.} \end{cases} \quad (30)$$

To determine g_Ψ in (1), we have to calculate $P_R^{(l)}$ and λ_l. For call classes associated with C_{BNC} and C_{DCC} which arrive according to a Poisson process, the rejection probability can be directly computed as $P_R^{(l)} = \sum_{(i,n)\in S} [1 - I_{\mathrm{th}}(i,n,l)]\pi_s(i,n)$.

For the overflow processes which arrive according to an MMPP, we have

$$P_R^{(l)} = \sum_{(i,n)\in S} \lambda_{hv}^{(n)} [1 - I_{\mathrm{th}}(i,n,l)] \pi_s(i,n) / \sum_{(i,n)\in S} \lambda_{hv}^{(n)} \pi_s(i,n) \quad (31)$$

Furthermore, overflows λ_l can be calculated as $\lambda_2 = E_n[\lambda_{nv}^{(n)}]$ and $\lambda_4 = E_n[\lambda_{hv}^{(n)}]$. We next propose an algorithm to find the optimal Ψ^* that minimizes g_Ψ.

Without loss of generality, we may sort the the labels of call classes such that if $l \leq l'$ then $C_R^{(l)} \leq C_R^{(l')}$. Then, Theorems 3 asserts that the optimal policy is an ordered monotonic threshold curve. For $\mathrm{Th}(l,n)$, this implies the following relations

$$\begin{aligned} \text{Monotonicity}: \ & \mathrm{Th}(l,n) \leq \mathrm{Th}(l,n-1) \\ \text{Ordered}: \ & \mathrm{Th}(l,n) \leq \mathrm{Th}(l+1,n). \end{aligned} \quad (32)$$

Let us define the fictitious call classes of 0 (lowest priority) and 5 (highest priority) to set boundary conditions. We have

$$\mathrm{Th}(L+1,n) = C_c - 1, \ \ \mathrm{Th}(0,n) = 0 \quad (33)$$

$$\mathrm{Th}(l,-1) = C_c - 1, \ \ \mathrm{Th}(l,M+1) = 0 \quad (34)$$

TABLE III
SIMULATION PARAMETERS.

Parameter	Value	Parameter	Value
C_c	50 calls	μ_c	0.01
C_w	20 calls	μ_w	0.01
λ_c	0.4 calls/sec	C_{BNC}	5
λ_w	0.2 calls/sec	C_{BNO}	10
η_{hcc}	5×10^{-3}	C_{DCC}	30
η_h^i	0.01	C_{DHO}	40

We propose an iterative algorithm that is referred to as Structured Coordinate Search Algorithm (SCSA). SCSA performs a local search in the state space confined by our structural results. Its is given in Algorithm 1. The loop invariant is that, at the start of every iteration, Ψ_r is an ordered monotonic threshold policy (OMTP). Then through greedy selection of the threshold level at each point, a new OMTP is found that results in a lower cost. After a finite number of steps the OMTP can not be improved any further and the algorithm converges. In the next section, the convergence and the quality of the policies generated by SCSA will be studied through numerical experiments.

Algorithm 1 Structured Coordinate Search Algorithm (SCSA)

```
1: Initialize $\Psi_0$
   such that: $\forall(l,n) \quad \text{Th}^{\Psi_0}(l,n) = C_c - 1$
   $r := 0$
2: $r = r + 1$, $\Psi_r = \Psi_{r-1}$
3: For $l := 1, \ldots, L$
     For $n := 0, \ldots, M$
       $b_l = \max\left[\text{Th}^{\Psi_r}(l-1,n), \text{Th}^{\Psi_{r-1}}(l,n+1)\right]$
       $b_u = \min\left[\text{Th}^{\Psi_{r-1}}(l+1,n), \text{Th}^{\Psi_r}(l,n-1)\right]$
       $\text{Th}^{\Psi_r}(l,n) = \underset{b_l \le t \le b_u}{\text{argmin}}\ g_{\Psi_r}[\text{Th}(l,n) = t]$
4: if $\Psi_r = \Psi_{r-1}$ then
5:     Return Policy $\Psi_r$
6: else
7:     Go to step 2
8: end if
```

VIII. NUMERICAL RESULTS

We conduct extensive simulation experiments to study the performance of structured admission control. For each set of system parameters, the simulation has two stages. First, SCSA, implemented in Matlab, is used to obtain an optimal policy. Then, this policy is fed into a discrete-event simulator, built in C++, to find the resultant QoS performance. For each data point in the following, a total of 2×10^5 seconds are simulated. All parameters take their default values shown in Table III unless otherwise stated. We consider the underlay mesh topology given in Fig. 1, consisting of four APs. We assume $\phi(n_1, n_2) = 1/6$ for every pair of adjacent APs.

A. Optimality of Structured Admission Control

We have conducted an extensive number of experiments to compare the result of SCSA with Value Iteration (VI) [13] under the approximate model of Section V. All threshold levels obtained by using SCSA and VI showed complete matching. These observations suggest that, for all practical parameter sets, the proposed structured algorithm can be considered as a highly efficient yet reliable method to perform CAC in HWNs. As an example, for the system model in Fig. 1, the computed threshold policy is given in Table IV. Intuitively, due to the smooth and monotonic nature of the optimal threshold curves proved in Theorems 1–3, SCSA gradually converges to the optimal policy, rather than sub-optimal solutions.

TABLE IV
THRESHOLD POLICY $\text{Th}(l,n)$

$\text{TH}(l,n)$	**n=0**	**1**	**2**	**3**	**4**
l=1	43	36	20	19	19
2	49	45	43	41	40
3	49	49	49	49	47
4	49	49	49	49	49

TABLE V
SCSA CONVERGENCE SPEED, COMPARED WITH VI.

	SCSA	VI	ES
M	Time (sec)	Time (sec)	Order
4	12	483	2^{200}
8	59	5685	2^{400}
16	307	$\gg$5685	2^{800}

B. Convergence Speed

The convergence speed, in CPU time, of SCSA is compared against VI in Table V under the approximate model of Section V. Also, the size of the state space of an exhaustive search (ES) for the optimal policy, which is of order $O(2^{M.C_c})$, is given in the last column. It can be seen that knowledge of the policy structure allows us to compute the optimal policy in a much shorter time than either VI or ES.

C. SCSA vs. Complete Sharing and Guard-Channel Policies

The Complete Sharing (CS) policy refers to the admission policy in which $\Psi(s) = 1$, regardless of the system state as long as it remains within system capacity boundaries. The Guard-Channel (GH) policy is a simplified form of SCSA where the threshold is estimated based on arrival and handover rates, regardless of the state of other parts of network. To compute the parameters of the GH policy we use the fact that the optimal threshold for calls of class l, if we have no observation of n, is the expected value of $Th(l,n)$ with regard to n, i.e., $Th(l) = E_n[Th(l,n)]$.

In Fig. 4 and 5, the performance g_Ψ of SCSA, CS, and GH policies are compared for two different loading conditions over a range of η_h. In Fig. 4, we have $\lambda_{nc} = 0.4$ and $\lambda_{nw} = 0.2$ and in Fig. 5, we have $\lambda_{nc} = 0.6$ and $\lambda_{nw} = 0.3$. Here, η_h is the intensity of mobility as it measures the rate of leaving an underlay AP due to handover.

These figures show that, even though SCSA operates over a reduced state space, it can significantly improve system performance. It can be seen that in all cases the SCSA policy result in a lower total cost. However, its effect become more significant when the system load increases. Comparison of GH and SCSA policies shows that partial observation of mesh status allows us to considerably improve the system performance.

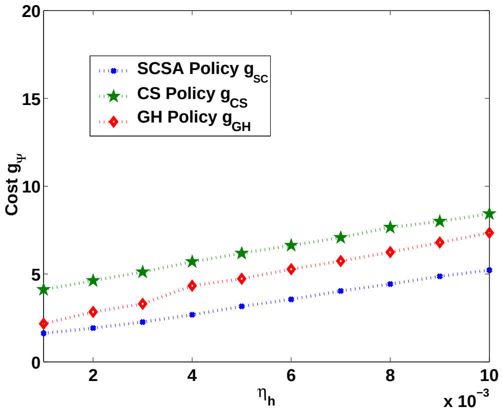

Fig. 4. Comparison of SCSA, CS and GH.

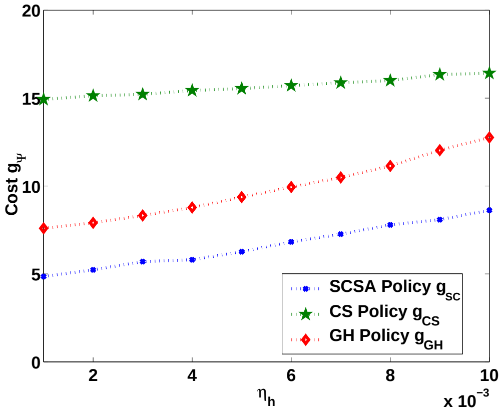

Fig. 5. Comparison of SCSA, CS and GH for increased load.

IX. Conclusion

In this paper, we characterize the structure of optimal admission control policies for Heterogeneous Wireless Networks (HWN), consisting of an integration of wireless mesh networks with an overlaying cellular infrastructure. A new bursty PO-MMPP traffic model is developed. This model captures the burstiness of the overflow traffic under the imperfect observability of the mesh network states.

For this system, structural optimal control results are presented. These results are used to design an efficient computational algorithm to determine the call admission policy. The performance of this algorithm is compared against the well-know Value Iteration technique, as well as the Complete Sharing and Guard-Channel policies. Discrete-event call-level simulations confirm that the obtained policy is effective in maintaining the desired QoS performance.

Acknowledgment

This work was supported in part by a grant from Bell Canada through its Bell University Laboratories R&D program.

References

[1] M. Stemm and R. H. Katz, “Vertical Handoffs in Wireless Overlay Networks,” *ACM/Springer Mobile Networks and Applications (MONET)*, vol. 3, no. 4, 1998.

[2] M. Buddhikot, G. Chandranmenon, S. Han, Y. Lee, S. Miller, and L. Salgarelli, “Integration of 802.11 and third-generation wireless data networks,” in *Proc. IEEE INFOCOM*, 2003, pp. 503– 512.

[3] S. Wu, K. Wong, and B. Li, “A dynamic call admission policy with precision QoS guarantee using stochastic control for mobile wireless networks,” *IEEE/ACM Trans. on Netw.*, vol. 10, no. 2, pp. 257–271, 2002.

[4] R. Ramjee, D. Towsley, and R. Nagarajan, “On optimal call admission control in cellular networks,” *Wireless Networks*, vol. 3, no. 1, pp. 29–41, 1997.

[5] M. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc. New York, NY, USA, 1994.

[6] M. Littman, T. Dean, and L. Kaelbling, “On the complexity of solving Markov decision problems,” in *Proceedings of the Eleventh Annual Conference on Uncertainty in Artificial Intelligence*, 1995, pp. 394–402.

[7] J. Ni, D. H. K. Tsang, S. Tatikonda, and B. Bensaou, “Optimal and Structured Call Admission Control Policies for Resource-Sharing Systems,” *IEEE Trans. on Comm.*, vol. 55, no. 1, pp. 158–170, 2007.

[8] E. Altman, T. Jimenez, and G. Koole, “On optimal call admission control in resource-sharing system,” *IEEE Transactions on Communications*, vol. 49, no. 9, pp. 1659–1668, 2001.

[9] K. Ross and D. Tsang, “The stochastic knapsack problem,” *IEEE Transactions on Communications*, vol. 37, no. 7, pp. 740–747, 1989.

[10] C. Boutilier, R. Dearden, and M. Goldszmidt, “Exploiting structure in policy construction,” in *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, 1995, pp. 1104–1111.

[11] A. Farbod and B. Liang, “Efficient structured policies for admission control in heterogeneous wireless networks,” *ACM/Springer Mobile Networks and Applications (MONET)*, vol. 12, no. 5, pp. 309–323, 2007.

[12] R. Guerin and L. Lien, “Overflow analysis for finite waiting room systems,” *IEEE Trans. on Comm.*, vol. 38, no. 9, pp. 1569–1577, 1990.

[13] S. M. Ross, *Applied Probability Models with Optimization Applications*. Dover Publications, 1992.

[14] W. Song and W. Zhuang, “QoS provisioning via admission control in cellular/wireless LAN interworking,” in *Proc. 2nd International Conference on Broadband Networks*, 2005, pp. 543– 550 Vol. 1.

[15] E.Stevens-Navarro, A. Rad, and V. Wong, “On Optimal Admission Control for Multi-Service Cellular/WLAN Interworking,” in *IEEE Global Tel. Conf. (GLOBECOM ’07)*, 2007, pp. 5042–5047.

[16] X. Lagrange and P. Godlewski, “Teletraffic analysis of a hierarchical cellular network,” in *Proc. IEEE 45th Veh. Tech. Conf.*, 1995.

[17] S. Chung and J. Lee, “Performance Analysis and Overflowed Traffic Characterization in Multiservice Hierarchical Wireless Networks,” *IEEE Trans. on Wireless Communications*, vol. 4, no. 3, pp. 904–918, 2005.

[18] L. Hu and S. Rappaport, “Personal communication systems using multiple hierarchical cellularoverlays,” *IEEE Journal on Selected Areas in Communications,*, vol. 13, no. 2, pp. 406–415, 1995.

[19] Q. Huang, K. Ko, and V. Iversen, “Approximation of loss calculation for hierarchical networks with multiservice overflows,” *IEEE Transactions on Communications*, vol. 56, no. 3, pp. 466–473, 2008.

[20] A. Kuczura, “The interrupted Poisson process as an overflow process,” *The Bell System Technical Journal*, vol. 52, no. 3, pp. 437–448, 1973.

[21] K. Meier-Hellstern, “The analysis of a queue arising in overflow models,” *IEEE Trans. on Comm.*, vol. 37, no. 4, pp. 367–372, 1989.

[22] F. Kelly, “Loss networks,” *Ann. Appl. Prob*, vol. 1, no. 3, pp. 319–378, 1991.

[23] K. Maheshwari and A. Kumar, “Performance analysis of microcellization for supporting twomobility classes in cellular wireless networks,” *IEEE Trans. on Vehicular Technology*, vol. 49, no 2, pp. 321–333, 2000.

[24] S. Wie, J. Jang, B. Shin, and D. Cho, “Handoff analysis of the hierarchical cellular system,” *IEEE Transactions on Vehicular Technology*, vol. 49, no. 5, pp. 2027–2036, 2000.

[25] Y. Fang, I. Chlamtac, and Y.-B. Lin, “Channel occupancy times and handoff rate for mobile computing and PCS networks,” *IEEE Transactions on Computers*, vol. 47, no. 6, pp. 679–692, 1998.

[26] V. Varma, S. Ramesh, K. Wong, M. Barton, G. Hayward, and J. Friedhoffer, “Mobility management in integrated UMTS/WLAN networks,” in *Proc. IEEE Int. Conf. on Comm.*, 2003, pp. 1048–1053.

[27] H. C. Tijms, *A First Course in Stochastic Models*. John Wiley and Sons Ltd., 2003.

[28] S. Stidham and R. Weber, “A survey of Markov decision models for control of networks of queues,” *Queueing Systems.*, vol. 13, pp. 291–314, 1993.

[29] B. Chang and R. Hwang, “Performance analysis for hierarchical multirate loss networks,” *IEEE/ACM Transactions on Networking*, vol. 12, no. 1, pp. 187–199, 2004.

[30] P. Courtois, “On time and space decomposition of complex structures,” *Communications of the ACM*, vol. 28, no. 6, pp. 590–603, 1985.

[31] E. Aarts and J. Lenstra, *Local Search in Combinatorial Optimization*. Princeton University Press, 2003.