ARTICLE IN PRESS

Performance Evaluation ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Performance Evaluation

journal homepage: www.elsevier.com/locate/peva

Behavioural equivalences and interference metrics for mobile ad-hoc networks

Michele Bugliesi [a], Lucia Gallina [a], Sardaouna Hamadou [b], Andrea Marin [a,*], Sabina Rossi [a]

[a] *Università Ca' Foscari Venezia, Italy*

[b] *Saclay and LIX, École Polytechnique, France*

ARTICLE INFO

Article history:
Available online xxxx

Keywords:
MANETs
Process algebra
Interference
Quantitative analysis

ABSTRACT

Connectivity and communication interference are two key aspects in mobile ad-hoc networks (MANETs). This paper proposes a process algebraic model targeted at the analysis of both such aspects. The framework includes a probabilistic process calculus and a suite of analytical techniques based on a probabilistic observational congruence and an interference-sensitive preorder. The former enables the verification of behavioural equivalences; the latter makes it possible to evaluate the interference level of behaviourally equivalent networks. The result is a comprehensive and effective framework for the behavioural analysis and a quantitative assessment of interference for wireless networks in the presence of node mobility. We show our techniques at work on two realistic case studies.

1. Introduction

Mobile ad-hoc networks are systems of mobile devices communicating over wireless links without a pre-established connectivity structure. Connectivity and communication interference are two key aspects in such networks. Node mobility is unconstrained: each device in a MANET moves autonomously, thereby seamlessly modifying the underlying topology, and hence creating the need for dynamic routing algorithms to ensure the desired level of connectivity among the mobile network nodes. Communication interference, in turn, is especially challenging in MANETs, as the half-duplex nature of wireless channels makes it impossible for a transmitter to automatically detect the presence of other, conflicting transmitters on the same channel. As a consequence, interfering transmissions may only be detected by receivers located at the intersection of the emitters' transmission ranges. The problem is even more complex in the presence of node mobility due to the dynamic structure of the network topology. While ad-hoc protocols that address these problems exist in the current literature [1,2], controlling interferences remains one of the pivotal aspects in the design of MANETs.

Drawing on earlier work on the subject (by the authors [3,4], and by others [5,6]), the present paper introduces a calculus to provide a formal basis for the analysis of connectivity and the evaluation of interference in MANETs. Like its predecessors [3,6], the new calculus is built around nodes, representing the devices of the systems, and locations, identifying the position cells across which each device may move inside the network. Node mobility is governed by probability distributions as in [3]. Conversely, wireless synchronization is non-deterministic, and controlled by (sequential) processes inside the nodes: each transmission broadcasts a message at a given radio frequency and within a given transmission range. Importantly, multiple

* Corresponding author. Tel.: +39 0412348476.
E-mail addresses: bugliesi@unive.it (M. Bugliesi), lgallina@dais.unive.it (L. Gallina), sardaouna.hamadou@polymtl.ca (S. Hamadou), marin@dsi.unive.it, marin@dais.unive.it (A. Marin), srossi@dais.unive.it (S. Rossi).

0166-5316/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.peva.2013.11.003

nodes may simultaneously transmit along the same channel, within overlapping areas: the calculus provides for an explicit representation of the collisions that may occur at the receiver sites.

The semantics of the calculus is inspired by Segala's probabilistic automata [7], and driven by schedulers to resolve the non-deterministic choice among the probability distributions over target states. We define a probabilistic observational congruence in the style of [8] to equate networks exhibiting the same observable behaviour. As in [4,3], and in contrast to [6], the notion of observability is associated with nodes listening at specific locations in the network, so as to allow a fine grained analysis of connectivity and interference at different network areas. We give a co-inductive characterization of observational congruence based on a labelled transition semantics. Then, we introduce interference-sensitive preorders over networks to measure the relative interference level of different, but observationally equivalent, networks.

The result is a comprehensive framework for the behavioural analysis and a quantitative assessment of interference for wireless networks in the presence of node mobility. We demonstrate the effectiveness of this framework on two cases studies. The first is an in-depth analysis of the well-known *Alternating Bit Protocol*, in which we contrast the standard implementation of the protocol with an alternative implementation that exploits an interference cancellation scheme for CDMA transmissions. Based on our framework, we are able to show that the two solutions are observationally equivalent, but the latter is superior as it guarantees a strictly lower level of interference. The second case study focuses on routing protocols with a comparative analysis of simple route discovery protocols based on AODV-like flooding policies [9] and *Location Aided Routing (LAR)* protocols [10] which try to control the flooding by addressing route requests to specific areas of the network based on information about the nodes' locations. We show that the LAR heuristic is equally effective in path discovery with respect to the flooding algorithms.

Related Work. The analysis of mobile and sensor networks has attracted broad interest in the literature on process algebraic and probabilistic models.

Various proposals target the analysis of behavioural properties related to node mobility, network connectivity, communication and message routing in non-deterministic settings. Merro introduces CMN [6], a *value-passing* CCS style [11] calculus with nodes and locations which has inspired a number of the initial design choices of our calculus. Singh, Ramakrishnan and Smolka define the ω-calculus [12], a conservative extension of the π-calculus which combines node mobility, and various forms of communication with the pi-calculus native mechanisms of scope extrusion by which nodes may also be connected with private channels. Nanz and Hankin introduce CBS$^{\#}$ [13], an extension of the *Calculus of Broadcasting Systems* [14]: their mechanisms for communication is related to ours in that transmissions are not atomic (when a node executes an output the topology of the network may change arbitrarily before the reception of the message by the neighbours of the sender). Sangiorgi and Lanese also address non-atomic transmissions in their calculus CWS [5], and specifically target a detailed analysis of interferences. Their framework, however, does not include node mobility nor do they introduce any interference metric. van Glabbeek et al. proposes AWN [15], a process algebra equipped with communication mechanisms and data structures specifically targeted at very precise and detailed modelling of wireless mesh routing protocols.

Several other papers propose probabilistic and stochastic models to provide quantitative analysis for various purposes. Song and Godskesen [16] propose a probabilistic broadcast calculus for mobile and wireless networks with unreliable connections. They do no address interference, and focus instead on message loss which in their calculus may only arise with a certain inherent probability and as a consequence of change in connectivity, determined by mobility (which, in turn, is governed by probabilities). Palamidessi et al. [17] define an extension of the applied pi-calculus with non-deterministic and probabilistic choice operators: our notion of probabilistic observational congruence is directly inherited from their work. Merro et al. discuss TCWS [18], a timed broadcasting process calculus targeted at security analysis of wireless networks with fixed nodes communicating at the same transmission power and over the same transmission frequency. Lanotte and Merro propose a probabilistic version of TCWS [19], aimed at the analysis of communication protocols. The main peculiarity of this calculus is the definition of a relation of *simulation up to probability*, which allows one to compare networks which exhibit the same behaviour up to a certain probability. This is an interesting result with respect, e.g., to the probabilistic applied π-calculus, presented in [17], where two networks can be compared only if they have exactly the same probability of performing observable actions. On the other hand, their model inherits the limitations due to the absence of mobility and of multiple frequencies of the original proposal in [18]. Hennessy and Cerone [20] propose a calculus to model the high-level behaviour of Wireless Systems (i.e., MAC-layer protocols). The calculus is characterized by a two-level structure: on one hand, it includes probabilistic and non-deterministic processes behaviour, as well as communications through a fixed set of channels; on the other hand, the topology is expressed through an undirected graph where each edge represents the direct link between a pair of network nodes. There is no notion of distance, nor of transmission radius; furthermore, modelling communication links with an undirected graph presupposes that all nodes use the same fixed radius to communicate, an assumption that is not realistic for MANETs, which include different kinds of devices, with different physical structure and power resources.

In the context of performance evaluation, Hillston [21] introduces the Performance Evaluation Process Algebra (PEPA) which is used for modelling systems composed of concurrently active components which co-operate and share work. Bernardo et al. introduce the Extended Markovian Process Algebra (EMPA$_{gr}$) [22]. All those calculi are built upon atomic actions and do not allow multiple devices to transmit at the same time. Although these shortcomings are overcome by the Hermanns' Interactive Markov Chains (IMCs) [23], the process algebra we propose deals both with non-determinism and probabilistic behaviours. This allows us to naturally model node mobility, transmission interferences and define

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

Table 1
Syntax.

Networks			Processes
$M, N ::= \mathbf{0}$	Empty network	$P, Q ::= \mathbf{0}$	Inactive process
$\mid n[P]_l$	Node (or device)	$\mid \mathtt{in}(c, \tilde{x}).P$	Input
$\mid (\nu c)M$	Channel restriction	$\mid \mathtt{out}\langle c_{L,r}, \tilde{w}\rangle.P$	Output
$\mid M_1 \mid M_2$	Parallel composition	$\mid [w_1 = w_2]P, Q$	Matching
		$\mid A\langle \tilde{w}\rangle$	Recursion

observational relations aimed at capturing the peculiar aspects of ad-hoc wireless networks. In our model, the time is partially abstracted out and we leave to the schedulers the role of solving the non-determinism rather than using a semantics based on the activity durations.

Finally, existing frameworks based on Petri Nets and queueing networks fall short of accounting for node mobility while maintaining a good accuracy in specifying the protocol design [24,25].

Plan of the paper. Section 2 introduces the calculus and its observational semantics. Section 3 defines the LTS semantics and the associated notion of probabilistic bisimilarity. Section 4 develops a technique for measuring the level of interference. Sections 5 and 6 shows our framework at work on the two case studies. Section 7 concludes the paper.

The present paper is a revised version of [26], extended with proofs for all results, a new case study (in Section 6) and an extended review of related work.

2. The calculus

The calculus extends the Probabilistic Energy-aware Broadcast Unicast and Multicast (PEBUM) calculus introduced in [3] with a new semantics of communication. The novelty of the present extension is the non-atomicity of the output and input actions, which we define after [5] to capture the presence of interference caused by the simultaneous transmissions of two (or more) nodes using the same channel in a common transmission area.

We use letters c for *channels*, n for *nodes*, l for *locations*, r for *transmission radii*, x and y for *variables*. *Closed values* contain nodes, locations, transmission radii and any basic value (booleans, integers, ...). *Values* include also variables. We use u and v for closed values and w for (open) values, and write $\tilde{v}$, $\tilde{w}$ for tuples of values, $\mathcal{N}$ for the set of networks, $\mathbf{C}$ for the set of channels and **Loc** for the set of all locations. As anticipated, while movements may be assumed to be continuous, we identify locations as the countable set of cells that constitute the observing areas within the network. The syntax of our calculus is shown in Table 1.

Networks are collections of nodes, devices that run concurrently and use channels to exchange messages. $\mathbf{0}$ denotes the empty network and $M_1|M_2$ the parallel composition of two networks. $n[P]_l$ is a network node named n located at the physical location l, and executing the process P. In $(\nu c)M$ the channel c is private with scope M, and we say it is bound in M: we denote by $fc(M)$ the set of channels which are not bound in M. We remark that in our calculus channels are distinct from values and cannot be transmitted; furthermore, given the structure of the syntactic productions, channels may not be dynamically created and thus $(\nu c)M$ simply plays the role of a CCS-style hiding operator.[1]

Processes are sequential and run inside nodes: $\mathbf{0}$ is the inactive process; $\mathtt{in}(c, \tilde{x}).P$ is ready to listen to a transmission, while $\mathtt{out}\langle c_{L,r}, \tilde{w}\rangle.P$ is ready to transmit. In $\mathtt{in}(c, \tilde{x}).P$, the variables in $\tilde{x}$ are bound with scope in P. In $\mathtt{out}\langle c_{L,r}, \tilde{w}\rangle.P$, the tag r represents the transmission radius of the sender: the choice of specific transmission ranges may depend on various parameters, and is left to the process running inside the transmitter node. The tag L, in turn, signals the locations from which the transmission will be observed.

The remaining syntactic forms are: $[w_1 = w_2]P, Q$ behaves as P if $w_1 = w_2$, and as Q otherwise. $A\langle \tilde{w}\rangle$ is the process defined via a (possibly recursive) definition $A(\tilde{x}) \stackrel{\text{def}}{=} P$, with $|\tilde{x}| = |\tilde{w}|$ where $\tilde{x}$ contains all channels and variables that are free in P.

Two further process forms arise as a result of reduction. In particular, processes that are ready to send or receive evolve into active senders and receivers:

P, Q	$::=$	...	The expressions of Table 1
	$\mid$	$c(\tilde{x}).P$	Active input
	$\mid$	$\bar{c}_{L,r}\langle \tilde{w}\rangle.P$	Active output.

Here, $c(\tilde{x}).P$ is actively receiving a tuple $\tilde{w}$ of (closed) values via channel c and continues as $P\{\tilde{w}/\tilde{x}\}$, i.e., as P with $\tilde{w}$ substituted for $\tilde{x}$ (where $|\tilde{x}| = |\tilde{w}|$). Dually, $\bar{c}_{L,r}\langle \tilde{w}\rangle.P$ is transmitting a tuple of values $\tilde{w}$ via channel c and then continues as P. Processes of the form $c(\tilde{x}).P$ or $\bar{c}_{L,r}\langle \tilde{w}\rangle.P$ are called *active*. Predicate $\mathtt{Active}(P)$ is $\mathtt{true}$ when P is active, and $\mathtt{A}(M)$ denotes the network composed of all the active nodes in M, i.e., all nodes $n[P]_l$ in M with P active.

[1] Since channels represent radio frequencies, they are all public, hence they need not be transmitted, and may not be hidden in practice. Indeed, the use of the hiding operator is only meant to specialize the verification method to some specific class of contexts.

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

Node connectivity is verified by looking at the physical location and the transmission radius of the sender: a message broadcast by a node is received only by the nodes that lie in the area delimited by the transmission radius of the sender. We presuppose a function $d(\cdot,\cdot)$ which takes two locations and returns the distance separating them (function d can be simply the euclidean distance between two locations, or a more complex function dealing with potential obstacles).

A network M is defined as the parallel composition of nodes with pairwise-distinct names moving independently from each other. We denote by $\prod_{i\in I} M_i$ the parallel composition of the networks M_i, for $i \in I$. Each node n is associated with a pair $\langle r_n, \mathbf{J}^n\rangle$, where r_n is a non-negative real number denoting the maximum transmission radius that n can use to transmit, while $\mathbf{J}^n$ is the transition matrix of a discrete time Markov chain: each entry $\mathbf{J}^n_{lk}$ denotes the probability that the node n located at l may move to the location k. Hence, $\sum_{k\in\mathbf{Loc}} \mathbf{J}^n_{lk} = 1$ for all locations $l \in \mathbf{Loc}$. Static nodes are associated with the identity Markov chain, i.e., the identity matrix $\mathbf{J}^n_{ll} = 1$ for all $l \in \mathbf{Loc}$ and $\mathbf{J}^n_{lk} = 0$ for all $l \neq k$. We note by μ^n_l the probability distribution associated with node n located at l, that is, the function over $\mathbf{Loc}$ such that $\mu^n_l(k) = \mathbf{J}^n_{lk}$, for all $k \in \mathbf{Loc}$.[2]

Let n be a node of a network M and l its location. In the following we denote by $M\{n : l'/l\}$ the network obtained by substituting l by l' inside the node n and by $[\![M]\!]_{\mu^n_l}$ the probability distribution over the set of networks induced by μ^n_l and defined as follows: for all network M',

$$[\![M]\!]_{\mu^n_l}(M') = \begin{cases} \mu^n_l(l') & \text{if } M' = M\{n : l'/l\} \\ 0 & \text{otherwise.} \end{cases}$$

Intuitively, $[\![M]\!]_{\mu^n_l}(M')$ is the probability that the network M evolves to M' due to the movement of its node n located at l. We say that M' is in the support of $[\![M]\!]_{\mu^n_l}$ if $[\![M]\!]_{\mu^n_l}(M') \neq 0$. We write $[\![M]\!]_{\Delta}$ for the Dirac distribution on the network M, namely the probability distribution defined as: $[\![M]\!]_{\Delta}(M) = 1$ and $[\![M]\!]_{\Delta}(M') = 0$ for all M' such that $M' \neq M$. Finally, we let θ range over $\{\mu^n_l \mid n \text{ is a node and } l \in \mathbf{Loc}\} \cup \{\Delta\}$.

Example 2.1 (*Probability Distributions*)**.** Consider a network

$$M = n_1[\mathtt{out}\langle c_{L,r_1}, \tilde{v_1}\rangle.P_1]_{l_1} \mid n_2[\mathtt{out}\langle c_{L,r_2}, \tilde{v_2}\rangle.P_2]_{l_2} \mid m[\mathtt{in}(c, \tilde{x}).P_3]_k$$

consisting of two mobile sender nodes, n_1 and n_2, communicating with a static receiver node m. Node n_1 moves back and forth between locations l_1 and l_2 according to the probability distribution defined by the discrete time Markov chain with the following transition matrix

$$\mathbf{J} = \begin{vmatrix} 1-p & p \\ q & 1-q, \end{vmatrix}$$

where $0 < p, q < 1$. Similarly, n_2 moves between l_2 and l_1 according to the same transition matrix $\mathbf{J}$. Then the probabilistic mobility of the network induced by the movement of the node n_1 is

$$[\![M]\!]_{\mu^{n_1}_{l_1}}(M') = \begin{cases} 1-p & \text{if } M' = M\{n_1 : l_1/l_1\} = M \\ p & \text{if } M' = M\{n_1 : l_2/l_1\} \\ 0 & \text{otherwise.} \end{cases}$$

Similarly for the second node we have

$$[\![M]\!]_{\mu^{n_2}_{l_2}}(M') = \begin{cases} 1-q & \text{if } M' = M\{n_2 : l_2/l_2\} = M \\ q & \text{if } M' = M\{n_2 : l_1/l_2\} \\ 0 & \text{otherwise} \end{cases}$$

while for the static receiver we have

$$[\![M]\!]_{\mu^m_k}(M') = \begin{cases} 1 & \text{if } M' = M\{m : k/k\} = M \\ 0 & \text{otherwise.} \end{cases}$$

Note that for the static node movement, we have $[\![M]\!]_{\mu^m_k} = [\![M]\!]_{\Delta}$.

The dynamics of the calculus is specified by the *probabilistic reduction relation* ($\rightarrow$), described in Table 3. It relies on an auxiliary relation, called structural congruence ($\equiv$), which is the least contextual equivalence relation satisfying the rules defined in Table 2. The probabilistic reduction relation takes the form $M \rightarrow [\![M']\!]_{\theta}$ denoting a transition that leaves from network M and leads to a probability distribution $[\![M']\!]_{\theta}$.

The synchronization over a wireless channel is described by the two rules (R-Bgn-Bcast) and (R-End-Bcast). (R-Bgn-Bcast) models the start of a transmission, with node n transiting from ready to active state to transmit message $\tilde{v}$ on channel c with

[2] Notice that $\mathbf{J}^n$ is a matrix, while μ^n_l is a function. We also remark that when the set of locations is infinite, the transition matrix is infinite. There are indeed possible situations: (i) the set of locations is infinite but each node moves only in a finite portion or, (ii) the locations reachable from a node also are infinitely many. In the first case the model is tractable with a sparse representation; in the second case, we may resort to the common assumption that transition matrix associated with the Markov chain has a regular block structure, hence admits a finite representation.

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

Table 2
Structural congruence.

$n[\mathbf{0}]_l \equiv \mathbf{0}$	(Struct Zero)
$n[[v = v]P, Q]_l \equiv n[P]_l$	(Struct Then)
$n[[v_1 = v_2]P, Q]_l \equiv n[Q]_l \; v_1 \neq v_2$	(Struct Else)
$n[A\langle\tilde{v}\rangle]_l \equiv n[P\{\tilde{v}/\tilde{x}\}]_l$ if $A(\tilde{x}) \stackrel{\text{def}}{=} P \wedge |\tilde{x}| = |\tilde{v}|$	(Struct Rec)
$M \mid N \equiv N \mid M$	(Struct Par Comm)
$(M \mid N) \mid M' \equiv M \mid (N \mid M')$	(Struct Par Assoc)
$M \mid \mathbf{0} \equiv M$	(Struct Zero Par)
$(\nu c)\mathbf{0} \equiv \mathbf{0}$	(Struct Zero Res)
$(\nu c)(\nu d)M \equiv (\nu d)(\nu c)M$	(Struct Res Res)
$(\nu c)(M \mid N) \equiv M \mid (\nu c)N$ if $c \notin fc(M)$	(Struct Res Par)

Table 3
Reduction semantics.

(R-Bgn-Bcast)
$$\frac{\forall i \in I.d(l, l_i) > r_i \;\; \forall i \in I \; \forall j \in J.d(l_i, l_j) > r_i \;\; \forall h \in (J \cup K).d(l, l_h) \leq r}{n[\mathtt{out}\langle c_{L,r}, \tilde{v}\rangle.P]_l \mid M \rightarrow [\![n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid M']\!]_\Delta}$$
where
$M \equiv \prod_{i\in I} n_i[\bar{c}_{L_i,r_i}\langle\tilde{v}_i\rangle.P_i]_{l_i} \mid \prod_{j\in J} n_j[\mathtt{in}(c, \tilde{x}_j).P_j]_{l_j} \mid \prod_{k\in K} n_k[c(\tilde{x}_k).P_k]_{l_k}$,
$M' \equiv \prod_{i\in I} n_i[\bar{c}_{L_i,r_i}\langle\tilde{v}_i\rangle.P_i]_{l_i} \mid \prod_{j\in J} n_j[c(\tilde{x}_j).P_j]_{l_j} \mid \prod_{k\in K} n_k[P_k\{\perp/\tilde{x}_i\}]_{l_k}$

(R-End-Bcast)
$$\frac{\forall j \in J.d(l, l_j) \leq r}{n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid \prod_{j\in J} n_j[c(\tilde{x}_j).P_j]_{l_j} \rightarrow [\![n[P]_l \mid \prod_{j\in J} n_j[P_j\{\tilde{v}/\tilde{x}_j\}]_{l_j}]\!]_\Delta}$$

(R-Res) $\dfrac{M \rightarrow [\![M']\!]_\theta}{(\nu c)M \rightarrow [\![(\nu c)M']\!]_\theta}$ (R-Move) $\dfrac{\mathtt{Active}(P) = \mathtt{false}}{n[P]_l \rightarrow [\![n[P]_l]\!]_{\mu_l^n}}$

(R-Par) $\dfrac{M \rightarrow [\![M']\!]_\theta}{M \mid N \rightarrow [\![M' \mid N]\!]_\theta}$ (R-Struct) $\dfrac{N \equiv M \;\; M \rightarrow [\![M']\!]_\theta \;\; M' \equiv N'}{N \rightarrow [\![N']\!]_\theta}$

radius r. The state change in n may cause a collision, which the rule captures as follows. We abuse the notation and write $n_h \in H$ to note nodes n_h with $h \in H$, for any index set H. The premise of the rule describes a situation in which nodes $n_k \in K$ and $n_i \in I$ are actively involved in a synchronization, while node n and the $n_j \in J$ are in (output and input, respectively) ready state. Given that all the active transmitters are out of n's range (because $d(l, l_i) > r_i$), n transits into active state: this awakes the $n_j \in J$, as they are now in range of an active transmitter, and at the same time causes a collision at the $n_k \in K$, which also are in range and were already active on input: as a result the $n_k \in K$ exit their active state, receiving the error signal $\perp$. All the remaining active receivers that do not sense a collision, and are in the range of an active sender may conclude the synchronization (see the R-End-Bcast rule).

As we mentioned earlier, the label L signals the set of locations at which the transmission will be observed. Notice that L does not play a role in a synchronization reduction, as messages are broadcast and observable (and received) by any active receiver in range. On the other hand, we use L to fine-tune our notion of observation in the definition of barb, to be discussed shortly.

Example 2.2 (*Interference*)**.** Consider again the network of the previous example where the two sender nodes are not within the radius of each other, i.e., $d(l_1, l_2) > \max(r_1, r_2)$, and they are both able to reach the receiver, i.e., $d(l_1, k) \leq r_1$ and $d(l_2, k) \leq r_2$. Then the following reductions, obtained by applying rule (R-Bgn-Bcast), lead to a state where an interference is caused at the receiver node:

$$M \rightarrow [\![n_1[\bar{c}_{L,r_1}\langle\tilde{v_1}\rangle.P_1]_{l_1} \mid n_2[\mathtt{out}\langle c_{L,r_2}, \tilde{v_2}\rangle.P_2]_{l_2} \mid m[c(\tilde{x}).P_3]_k]\!]_\Delta$$

and if $M' = n_1[\bar{c}_{L,r_1}\langle\tilde{v_1}\rangle.P_1]_{l_1} \mid n_2[\mathtt{out}\langle c_{L,r_2}, \tilde{v_2}\rangle.P_2]_{l_2} \mid m[c(\tilde{x}).P_3]_k$ then

$$M' \rightarrow [\![n_1[\bar{c}_{L,r_1}\langle\tilde{v_1}\rangle.P_1]_{l_1} \mid n_2[\bar{c}_{L,r_2}\langle\tilde{v_2}\rangle.P_2]_{l_2} \mid m[P_3\{\perp/\tilde{x}\}]_k]\!]_\Delta .$$

The first sender node starts broadcasting on the channel c causing the receiver to become active. Then the second sender being too far away from n_1 to notice that the channel is occupied starts broadcasting on the same channel and hence causes an interference at the receiver side. If we are interested in observing the transmissions at location k, i.e., $k \in L$ then our semantics will allows us to detect the interference.

Rule (R-Move) describes node mobility. A node n located at l and executing a move action will reach a location with a probability described by the distribution μ_l^n that depends on the Markov chain $\mathbf{J}^n$ statically associated with n. We assume that a node can move only if it is not actively involved in any synchronization: as a result, nodes may move before starting a synchronization (when they are in a ready, but not active state), while they are static during the actual synchronization. This is a reasonable assumption in wireless network analysis, since, in most practical situations, packet transmission delays may be assumed to be orders of magnitude faster than node mobility.

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

All the remaining rules are standard [11], but a further remark is in order about the (R-Par) rule and its interaction with the rules that govern synchronization. In fact, such interactions may give rise to inconsistent network configurations. To see that, observe that an application of the (R-Par) rule may cause messages to be lost by active receivers located within the range of an active sender, even when there is no interference. Similarly, an application of (R-Par) may exclude any set of active sender and/or receiver from a synchronization: in both cases, the network is left in an inconsistent state, with active senders (dually receivers) and no receiver (sender) in range.

Example 2.3 (*Inconsistent Networks*)**.** Consider again the network of the previous example where now the two sender nodes are within the radius of each other, that is $d(l_1, l_2) \leq \min(r_1, r_2)$. By applying rule (R-Bgn-Bcast) we obtain

$$M \to [\![n_1[\mathtt{out}\langle c_{L,r_1}, \tilde{v_1}\rangle.P_1]_{l_1} \mid n_2[\bar{c}_{L,r_2}\langle \tilde{v_2}\rangle.P_2]_{l_2} \mid m[c(\tilde{x}).P_3]_k]\!]_{\Delta} .$$

Now let $M' = n_1[\mathtt{out}\langle c_{L,r_1}, \tilde{v_1}\rangle.P_1]_{l_1} \mid n_2[\bar{c}_{L,r_2}\langle \tilde{v_2}\rangle.P_2]_{l_2} \mid m[c(\tilde{x}).P_3]_k$. The following reduction obtained by applying rule (R-Par)

$$M' \to [\![n_1[\bar{c}_{L,r_1}\langle \tilde{v_1}\rangle.P_1]_{l_1} \mid n_2[\bar{c}_{L,r_2}\langle \tilde{v_2}\rangle.P_2]_{l_2} \mid m[c(\tilde{x}).P_3]_k]\!]_{\Delta}$$

leads to an inconsistent state where both sender nodes are broadcasting on the same channel while being within a reachable distance of each other. Similarly, consider the following application of rule (R-Bng-Bcast):

$$M \to [\![n_1[\bar{c}_{L,r_1}\langle \tilde{v_1}\rangle.P_1]_{l_1} \mid n_2[\mathtt{out}\langle c_{L,r_2}, \tilde{v_2}\rangle.P_2]_{l_2} \mid m[c(\tilde{x}).P_3]_k]\!]_{\Delta} .$$

If $M'' = n_1[\bar{c}_{L,r_1}\langle \tilde{v_1}\rangle.P_1]_{l_1} \mid n_2[\mathtt{out}\langle c_{L,r_2}, \tilde{v_2}\rangle.P_2]_{l_2} \mid m[c(\tilde{x}).P_3]_k$ then by an application of rule (R-Par) we obtain

$$M'' \to [\![n_1[P_1]_{l_1} \mid n_2[\mathtt{out}\langle c_{L,r_2}, \tilde{v_2}\rangle.P_2]_{l_2} \mid m[c(\tilde{x}).P_3]_k]\!]_{\Delta}$$

leading to an inconsistent state where m is actively receiving a message while there is no active sender.

While it would be possible to rectify the problem by including conditions to exclude critical pairs for the (R-Par) and the synchronization rules, it is technically more convenient to simply disregard any undesired reduction. This is achieved in our framework by resorting to the notion of "admissible scheduler" (discussed shortly) to guide the dynamics of networks through "well-formed" executions.

Formally, given a network M, we write $M \to_\theta N$ if $M \to [\![M']\!]_\theta$ and N is in the support of $[\![M']\!]_\theta$. Following [17], an execution for M is a (possibly infinite) sequence of steps $M \to_{\theta_1} M_1 \to_{\theta_2} M_2 \ldots$. We write $Exec_M$ for the set of all possible executions starting from M, $last(e)$ for the final state of a *finite* execution e, e^j for the prefix execution $M \to_{\theta_1} M_1 \cdots \to_{\theta_j} M_j$ of length j of the execution $e = M \to_{\theta_1} M_1 \cdots \to_{\theta_j} M_j \to_{\theta_{j+1}} M_{j+1} \cdots$, and $e\uparrow$ for the set of e' such that e is a prefix of e'. We write $M \to^* M'$ if there exists a finite execution $e \in Exec_M$ such that $last(e) = M'$.

Following [8], we formalize the observational semantics for our calculus in terms of a notion of *barb* that provides the basic unit of observation. As in other calculi for wireless communication [6,27], the definition of barb is naturally expressed in terms of message transmission.

We denote by $behave(M) = \{[\![M']\!]_\theta \mid M \to [\![M']\!]_\theta\}$ the set of the possible behaviours of M. In order to solve the non-determinism in a network execution, we consider each possible probabilistic transition $M \to [\![M']\!]_\theta$ as arising from a *scheduler* (see [7,17]). Let $Exec^f$ be the set of all finite executions and $Behave(Exec^f)$ the set of all the distributions in $behave(last(e))$ with $e \in Exec^f$. Then, a *scheduler* is a total function F from $Exec^f$ to $Behave(Exec^f)$ assigning to a finite execution e a distribution $[\![N]\!]_\theta \in behave(last(e))$. Notice that we consider *deterministic* schedulers in the style of [17] rather than *randomized* ones as in [7]. Indeed, we aim at modelling network behaviours where probabilities are used to describe only node mobility while leaving the control of the transmissions to the standard deterministic scheduler. We define the set of executions starting from a network M and driven by a scheduler F as:

$$Exec^F_M = \{e = M \to_{\theta_1} M_1 \to_{\theta_2} M_2 \ldots \mid \forall j,\ M_{j-1} \to [\![M'_j]\!]_{\theta_j},\ [\![M'_j]\!]_{\theta_j} = F(e^{j-1}) \text{ and } M_j \text{ is in the support of } [\![M'_j]\!]_{\theta_j}\}.$$

Given a finite execution $e = M \to_{\theta_1} M_1 \cdots \to_{\theta_k} M_k$ starting from M and driven by a scheduler F we define

$$P^F_M(e) = [\![M'_1]\!]_{\theta_1}(M_1) \cdot \ldots \cdot [\![M'_k]\!]_{\theta_k}(M_k)$$

where $\forall j \leq k,\ [\![M'_j]\!]_{\theta_j} = F(e^{j-1})$. We define the probability space on the executions starting from a given network M as follows. Given a scheduler F, $\sigma Field^F_M$ is the smallest sigma field on $Exec^F_M$ that contains the basic cylinders $e\uparrow$, where $e \in Exec^F_M$. The probability measure $Prob^F_M$ is the unique measure on $\sigma Field^F_M$ such that $Prob^F_M(e\uparrow) = P^F_M(e)$. Given a measurable set of networks H, we note by $Exec^F_M(H)$ the set of executions starting from M and crossing a state in H. Formally $Exec^F_M(H) = \{e \in Exec^F_M \mid last(e^j) \in H \text{ for some } j\}$. We denote the probability for a network M to evolve into a network in H according to the policy given by F as $Prob^F_M(H) = Prob^F_M(Exec^F_M(H))$.

As anticipated, we restrict to suitable subclasses of networks and executions, namely *well-formed* networks and executions driven by *admissible* schedulers, respectively. Well formed-networks are such that (1) before transiting to active

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

state, each transmitter checks (locally) that the communication channel is not busy with other transmissions, and (2) each active receiver in the network is in the transmission cell of exactly one transmitter. Below we give the formal definition.

We recall the reader that $\mathtt{A}(M)$ is the network composed by active nodes in M and introduce the auxiliary operator $\mathtt{Top}(\cdot)$ over networks, used as follows: a channel c is at the top level of a network M, denoted $c \in \mathtt{Top}(M)$, if $M \equiv (\nu \tilde{d})(n[P]_l \mid N)$ and P is of the form $c(\tilde{x}).Q$ or $\bar{c}_{L,r}\langle \tilde{w} \rangle.Q$.

Definition 2.4 (*Well-formed Network*)**.** A network M is *well-formed* if either $\mathtt{A}(M) \equiv \mathbf{0}$ or $\mathtt{A}(M) \equiv (\nu \tilde{d})\big(\prod_{i\in I} n_i[\bar{c}_{L_i,r_i}\langle \tilde{v}_i \rangle.P_i]_{l_i} \mid \prod_{j\in J} n_j[c(\tilde{x}_j).P_j]_{l_j} \mid \mathtt{A}(N)\big)$ for some N and the following conditions hold:

- $\forall i, i' \in I. d(l_i, l_{i'}) > max(r_i, r_{i'})$,
- $\forall j \in J. \exists! i \in I$ such that $d(l_i, l_j) \leq r_i$,
- $c \notin \mathtt{Top}(\mathtt{A}(N))$, and N is well-formed.

Back to Example 2.3, we see that the final states of the reductions are not well-formed. In the first case, the inconsistent state breaks the first well-formedness condition in Definition 2.4, since there are two active senders on the same channel within the radius of each other; the second inconsistent state, in turn, breaks the second well-formedness condition as there is no single active sender reaching the active receiver. Restricting to *admissible* schedulers rules out any unwanted transition and inconsistent state, preserving network well-formedness along execution.

Definition 2.5 (*Admissible Scheduler*)**.** A scheduler F is *admissible* if for all executions e and for all networks M in the support of $F(e)$, M is well-formed. We let *Sched* note the set of all admissible schedulers.

Schedulers constitute an essential feature for modelling communication protocols, as they provide freedom in modelling implementation and incomplete knowledge of a system. Therefore in introducing our notion of network equivalence (cf. Definition 2.12 below) we seek parametricity with respect to the schedulers driving execution, so as to provide corresponding flexibility in the analysis. In addition, as it is customary in process algebraic frameworks, we expect our equivalence to be a congruence (equivalently, contextual).

In order to define a congruence relation among networks, we have to select a set of schedulers guaranteeing that network behaviour is preserved when the network in included in any possible context. We henceforth define a context as a network term with a hole $[\cdot]$, defined by the following grammar:

$$\mathcal{C}[\cdot] ::= [\cdot] \mid [\cdot]|M \mid M|[\cdot] \mid (\nu c)[\cdot].$$

The following definition introduces a relation between the executions of a network M and those of the same network once embedded into a context.

Definition 2.6. Let M_0 and $O_0 \equiv M_0$ be networks, $F, F' \in$ *Sched* admissible schedulers, C_0 a context, and $e \in Exec^F_{M_0}$ and $e' \in Exec^{F'}_{C_0[O_0]}$ two executions such that:

$$\begin{aligned} e &= M_0 \rightarrow_{\theta_1} M_1 \rightarrow_{\theta_2} M_2 \cdots \rightarrow_{\theta_h} M_h \\ e' &= C_0[O_0] \rightarrow_{\theta'_1} C_1[O_1] \rightarrow_{\theta'_2} C_2[O_2] \cdots \rightarrow_{\theta'_k} C_k[O_k]. \end{aligned}$$

We say that e and e' have the same behaviour with respect to M_0 – written $e \sim_{M_0} e'$ – if there exists a monotonic surjective function f from $[0..k]$ to $[0..h]$ such that:

(i) $\forall i \in [0..k], O_i \equiv M_{f(i)}$
(ii) $\forall j \in [1..k], \theta'_j = \theta_{f(j)}$ when $M_{f(j-1)} \rightarrow_{\theta_{f(j)}} M_{f(j)}$.

The next definition helps formalize our notion of observational congruence. Intuitively it defines a set of schedulers $F^M_{\mathcal{C}}$ that depends on F and a network M, and which includes F and all the schedulers driving M in an arbitrary context. The schedulers in $F^M_{\mathcal{C}}$ are selected based on the way they drive the interactions between the contexts and M, so as to ensure that they preserve the behaviour of M according to F (and are otherwise unconstrained in their driving any context behaviour independent of M).

Definition 2.7. Given a network M and an admissible scheduler $F \in$ *Sched*, we define the set $F^M_{\mathcal{C}}$ as follows:

$$F^M_{\mathcal{C}} = \{F' \in \mathit{Sched} \mid \forall C[\cdot] \text{ context}, \forall e' \in Exec^{F'}_{C[M]} \text{ there exists } e \in Exec^F_M \text{ such that } e \sim_M e'\}.$$

We say that $F^M_{\mathcal{C}}$ is consistent with F if $F \in F^M_{\mathcal{C}}$. Hereafter, we consider only schedulers F ensuring that $F^M_{\mathcal{C}}$ is consistent.

Given a network M and $\mathcal{F} \subseteq$ *Sched*, we also define $\mathcal{F}^M_{\mathcal{C}} = \bigcup_{F\in\mathcal{F}} F^M_{\mathcal{C}}$.

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

Example 2.8. Let $M_0 \equiv m[\mathtt{out}\langle c_{L,r}, v\rangle.P]_l$ and $F \in \mathit{Sched}$ such that

$$M_0 \rightarrow_{\Delta} M_1 \rightarrow_{\Delta} M_2 \in \mathit{Exec}^F_{M_0},$$

with $M_1 \equiv m[\bar{c}_{L,r}\langle v\rangle.P]_l$ and $M_2 \equiv m[P]_l$.

Consider $N_0 \equiv n[\mathtt{in}(c, x).Q]_k$ such that $d(l, k) \leq r$. All the admissible schedulers allowing M_0 and N_0 to interact are candidate for being in $F_{\mathcal{C}}^{M_0}$. Indeed, consider $F_1 \in \mathit{Sched}$ such that, by applying rules (Struct-Bgn-Bcast) and (Struct-End-Bcast)

$$M_0 \mid N_0 \rightarrow_{\Delta} M_1 \mid N_1 \rightarrow_{\Delta} M_2 \mid N_2 \in \mathit{Exec}^{F_1}_{M_0|N_0}$$

with $N_1 \equiv n[c(x).Q]_k$ and $N_2 \equiv n[Q\{v/x\}]_k$, and consider also F_2 such that, by applying rule (R-Par)

$$M_0 \mid N_0 \rightarrow_{\Delta} M_1 \mid N_0 \rightarrow_{\Delta} M_2 \mid N_0 \in \mathit{Exec}^{F_2}_{M_0|N_0}.$$

Both F_1 and F_2 satisfy the properties of Definition 2.7 when considering the context $N_0|[\cdot]$.

Consider now the network L_0 defined as $L_0 \equiv \ell[\mathtt{out}\langle c_{L,r}, w\rangle.R]_j$ with $d(j, k) \leq r$. Consider also the admissible scheduler F_1 for the network $M_0|N_0$ and let F_3 be a scheduler for $L_0|M_0|N_0$ such that:

$$L_0|M_0|N_0 \rightarrow_{\Delta} L_0|M_1|N_1 \rightarrow_{\Delta} L_1|M_1|N_3 \in \mathit{Exec}^{F_3}_{L_0|M_0|N_0},$$

with $L_1 \equiv \ell[\bar{c}_{L,r}\langle w\rangle.R]_j$ and $N_3 \equiv n[Q\{\perp/x\}]_k$. Then, according to Definition 2.7, $F_3 \notin F_{1\mathcal{C}}^{M_0|N_0}$. Notice that this example shows that although the contexts and a network can interact, the class of interactions allowed by Definition 2.7 are not completely arbitrary. Indeed, the scheduler F which is initially selected for the network will have an important role in the definition of our observational equivalence since only the behaviours admitted by the schedulers in $F_{\mathcal{C}}^{M}$ will be considered for the proposed definition of equivalence.

We are now ready to discuss our notion of observation. We first introduce a notation for *strong barbs*: for any network M, we write $M\downarrow_{c@K}$ whenever $M \equiv (\nu\tilde{d})(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid M')$ with $c \notin \tilde{d}$, $K \subseteq L$, $K \neq \emptyset$ and for all k in K, $d(l, k) \leq r$. In other words, the strong barb $M\downarrow_{c@K}$ signals that an active transmission from c can be observed in M from some of the intended observation points in L for that transmission. This notion of strong barb generalizes the corresponding notion in related calculi, notably [6]: indeed, taking L to be **Loc** uniformly on all output prefixes, our definition coincides with that in [6].

Example 2.9. Consider the network M of Example 2.2. If $k \in L$ then $M\downarrow_{c@\{k\}}$ otherwise $M \not\downarrow_{c@\{k\}}$.

Definition 2.10 (*Probabilistic Barb*)**.** A well-formed network M has a *barb* with probability p on a channel c at locations in K according to the scheduler F, written $M\Downarrow^F_p c@K$, if $\mathit{Prob}^F_M(H) = p$ with $H = \{M' \mid M \rightarrow^* M'\downarrow_{c@K}\}$.

Intuitively, for a given network M and scheduler F, if $M\Downarrow^F_p c@K$ then p is the positive probability that M, driven by F, performs a transmission on channel c and at least one of the nodes in the intended observation locations is able to correctly listen to it.

In the following, we introduce a probabilistic observational congruence, in the style of [17], parametrically with respect to a set of schedulers.

Definition 2.11. Given a set $\mathcal{F} \in \mathit{Sched}$ of schedulers, and a relation $\mathcal{R}$ over networks:

- *Barb preservation.* $\mathcal{R}$ is *barb preserving* w.r.t. $\mathcal{F}$ if $M\mathcal{R}N$ and $M\Downarrow^F_p c@K$ for some $F \in \mathcal{F}_{\mathcal{C}}^M$ implies that there exists $F' \in \mathcal{F}_{\mathcal{C}}^N$ such that $N\Downarrow^{F'}_p c@K$.
- *Reduction closure.* $\mathcal{R}$ is *reduction closed* w.r.t. $\mathcal{F}$ if $M\mathcal{R}N$ implies that for all $F \in \mathcal{F}_{\mathcal{C}}^M$, there exists $F' \in \mathcal{F}_{\mathcal{C}}^N$ such that for all classes $\mathcal{C} \in \mathcal{N}/\mathcal{R}$, $\mathit{Prob}^F_M(\mathcal{C}) = \mathit{Prob}^{F'}_N(\mathcal{C})$.
- *Contextuality.* $\mathcal{R}$ is *contextual* if $M\mathcal{R}N$ implies that for every context $C[\cdot]$ such that $C[M]$ and $C[N]$ are well formed, it holds that $C[M]\ \mathcal{R}\ C[N]$.

Definition 2.12 (*Probabilistic Observational Congruence w.r.t. $\mathcal{F}$*)**.** Given a set $\mathcal{F}$ of schedulers, the *probabilistic observational congruence w.r.t. $\mathcal{F}$*, written $\cong^{\mathcal{F}}_p$, is the largest symmetric relation over networks which is reduction closed, barb preserving and contextual.

3. A bisimulation-based proof technique

We develop a co-inductive proof technique for the probabilistic observational congruence $\cong^{\mathcal{F}}_p$.

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

Table 4
LTS rules for processes.

$$\text{(Beg-Out)}\ \frac{-}{\mathtt{out}\langle c_{L,r}, \tilde{v}\rangle.P \xrightarrow{\tilde{c}_{L,r}} \tilde{c}_{L,r}\langle\tilde{v}\rangle.P} \qquad \text{(End-Out)}\ \frac{-}{\tilde{c}_{L,r}\langle\tilde{v}\rangle.P \xrightarrow{\tilde{c}_{L,r}\tilde{v}} P}$$

$$\text{(Beg-In)}\ \frac{-}{\mathtt{in}(c,\tilde{x}).P \xrightarrow{c} c(\tilde{x}).P} \qquad \text{(End-In)}\ \frac{-}{c(\tilde{x}).P \xrightarrow{c\vartheta} P\{\vartheta/\tilde{x}\}}$$

$$\text{(Then)}\ \frac{P \xrightarrow{\eta} P'}{[\tilde{v} = \tilde{v}]P, Q \xrightarrow{\eta} P'} \qquad \text{(Else)}\ \frac{Q \xrightarrow{\eta} Q' \quad \tilde{v_1} \neq \tilde{v_2}}{[\tilde{v_1} = \tilde{v_2}]P, Q \xrightarrow{\eta} Q'}$$

$$\text{(Rec)}\ \frac{P\{\tilde{v}/\tilde{x}\} \xrightarrow{\eta} P' \quad A(\tilde{x}) \stackrel{\text{def}}{=} P}{A\langle\tilde{v}\rangle \xrightarrow{\eta} P'}$$

Table 5
LTS rules for networks.

$$\text{(Beg-Snd)}\ \frac{P \xrightarrow{\tilde{c}_{L,r}} P'}{n[P]_l \xrightarrow{c_L![l,r]} [\![n[P']_l]\!]_\Delta} \qquad \text{(End-Snd)}\ \frac{P \xrightarrow{\tilde{c}_{L,r}\tilde{v}} P'}{n[P]_l \xrightarrow{c_L!\tilde{v}[l,r]} [\![n[P']_l]\!]_\Delta}$$

$$\text{(Beg-Rcv)}\ \frac{P \xrightarrow{c} P'}{n[P]_l \xrightarrow{c?@l} [\![n[P']_l]\!]_\Delta} \qquad \text{(End-Rcv)}\ \frac{P \xrightarrow{c\vartheta} P'}{n[P]_l \xrightarrow{c?\vartheta@l} [\![n[P']_l]\!]_\Delta}$$

$$\text{(Beg-Bcast)}\ \frac{M \xrightarrow{c_L![l,r]} [\![M']\!]_\Delta \ N \xrightarrow{c?@l'} [\![N']\!]_\Delta \quad d(l,l') \leq r \ \wedge\ \mathtt{A}^s_N(c,l) = \mathtt{A}^s_N(c,l') = \emptyset}{M|N \xrightarrow{c_L![l,r]} [\![M'|N']\!]_\Delta}$$

$$\text{(Coll-Bcast)}\ \frac{M \xrightarrow{c_L![l,r]} [\![M']\!]_\Delta \ N \xrightarrow{c?\perp@l'} [\![N']\!]_\Delta \quad d(l,l') \leq r \ \wedge\ \mathtt{A}^s_N(c,l) = \emptyset}{M|N \xrightarrow{c_L![l,r]} [\![M'|N']\!]_\Delta}$$

$$\text{(End-Bcast)}\ \frac{M \xrightarrow{c_L!\tilde{v}[l,r]} [\![M']\!]_\Delta \ N \xrightarrow{c?\tilde{v}@l'} [\![N']\!]_\Delta \quad d(l,l') \leq r}{M|N \xrightarrow{c_L!\tilde{v}[l,r]} [\![M'|N']\!]_\Delta}$$

$$\text{(Lose1)}\ \frac{M \xrightarrow{c_L![l,r]} [\![M']\!]_\Delta}{M \xrightarrow{\tau} [\![M']\!]_\Delta} \qquad \text{(Lose2)}\ \frac{M \xrightarrow{c_L!\tilde{v}[l,r]} [\![M']\!]_\Delta}{M \xrightarrow{\tau} [\![M']\!]_\Delta}$$

$$\text{(Move)}\ \frac{\mathtt{Active}(P) = \mathtt{false}}{n[P]_l \xrightarrow{\tau} [\![n[P]_l]\!]_{\mu_l^n}} \qquad \text{(Res)}\ \frac{M \xrightarrow{\gamma} [\![M']\!]_\theta \ \mathtt{Chan}(\gamma) \neq c}{(\nu c)M \xrightarrow{\gamma} [\![(\nu c)M']\!]_\theta}$$

$$\text{(Obs)}\ \frac{M \xrightarrow{c_L!\tilde{v}[l,r]} [\![M']\!]_\Delta \ R = \{l' :\ d(l,l') \leq r \ \wedge\ |\ \mathtt{A}^s_M(c,l')\ |= 1\}\ K \subseteq R \cap L, K \neq \emptyset}{M \xrightarrow{c!\tilde{v}@K \lhd R} [\![M']\!]_\Delta}$$

$$\text{(Par)}\ \frac{M \xrightarrow{\gamma} [\![M']\!]_\theta}{M|N \xrightarrow{\gamma} [\![M'|N]\!]_\theta}$$

3.1. Labelled transition semantics

As for its predecessor, we define a LTS semantics for our calculus, which is built upon two sets of rules: one for processes and one for networks. Table 4 presents the LTS rules for processes. Transitions are of the form $P \xrightarrow{\eta} P'$, where η ranges over input and output actions of the form:

$$\eta ::= c|c\vartheta|\tilde{c}_{L,r}|\tilde{c}_{L,r}\tilde{v} \quad \text{with } \vartheta ::= \tilde{v} \mid \perp.$$

Rules (Beg-Out) and (End-Out) model the beginning and the end of an output action. Rule (Beg-In) models a process beginning listening to a channel in order to receive a value. Rule (End-In) models either the correct reception of a message or the reception of a $\perp$ due to a collision. All the remaining rules are standard as in [11].

Table 5 presents the LTS rules for networks. The transitions are of the form $M \xrightarrow{\gamma} [\![M']\!]_\theta$, where M is a network, $[\![M']\!]_\theta$ is a distribution over networks, and γ ranges over the following labels:

$$\gamma ::= c?@l|c?\vartheta@l|c_L![l,r]|c_L!\tilde{v}[l,r]|c!\tilde{v}@K \lhd R|\tau.$$

We denote by $\mathtt{A}^s_M(c,l)$ the set of active senders of M on channel c reaching l, i.e., if $\mathtt{A}(M) \equiv (\nu\tilde{d})\big(\prod_{i\in I} n_i[\tilde{c}_{L_i,r_i}\langle\tilde{v}_i\rangle.P_i]_{l_i} \mid \prod_{j\in J} n_j[c(\tilde{x}_j).P_j]_{l_j} \mid N\big)$ and $c \notin \mathtt{Top}(N)$ then $\mathtt{A}^s_M(c,l) = \{n_i : i \in I,\ d(l,l_i) \leq r_i\}$.

Rules (Beg-Snd) and (End-Snd) model the transmission of a message $\tilde{v}$ through channel c with radius r to the set L of observers. Transmissions are non-atomic actions: indeed, since mobile ad-hoc networks are not controlled by any fixed infrastructure, we have to take into account the possibility for nodes to be not perfectly synchronized with each other. (Beg-Rcv) models the beginning of a message reception, while (End-Rcv) models both the successful reception of a message

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

or the reception of a failure message (denoted by $\perp$) due to an interference. Rule (Beg-Bcast) models the beginning of a broadcast message propagation: all the nodes lying within the transmission cell of the sender may begin to receive a message (regardless of the fact that they are in L). Rule (Coll-Bcast) models the collision occurred at the location of a receiver lying within the intersection of the transmission area of different nodes transmitting simultaneously through the same channel. Rule (End-Bcast) models the conclusion of a broadcast message propagation: all the nodes lying within the transmission cell of the sender will successfully receive a message. Rule (Obs) models the observability of a transmission: every transmission may be detected (and hence *observed*) by any recipient located within the transmission cell of one sender and outside the "interference area", that is the intersection of the transmission areas of the active senders of the network. The label $c!\tilde{v}@K \lhd R$ represents the transmission of the tuple $\tilde{v}$ of messages via c to the subset K of observers inside the reachable locations R within the transmission cell of the sender. Notice that collisions are not observable and only a correctly ended transmission may be observed. Rule (Move) models migration of a mobile node n from a location l to a location k according to the probability distribution μ_l^n, which depends on the Markov chain $\mathbf{J}^n$ statically associated with n. Nodes can move only if they are not executing any active action (i.e., nodes cannot move while transmitting or receiving). Rules (Lose1) and (Lose2) model both message loss and a local activity of the network which an observer is not party to. As usual [11], τ-transitions are used to denote non-observable actions. Finally, rule (Res) models the standard channel restriction, where $\mathtt{Chan}(\gamma) = c$ if γ is of the form: $c?@l$; $c?\vartheta@l$; $c_L![l, r]$; $c_L!\tilde{v}[l, r]$ or $c!\tilde{v}@K \lhd R$, and $\mathtt{Chan}(\tau) = \perp$. Rule (Par) is defined as in [11].

We prove that the LTS-based semantics coincides with the reduction semantics and the notion of observability (barb) given in the previous section.

We first prove that if $M \xrightarrow{\gamma} [\![M']\!]_\Delta$, then the structure of M and M' can be determined up to structural congruence.

Lemma 3.1. *Let M be a network.*

1. *If $M \xrightarrow{c?@l} [\![M']\!]_\Delta$, then there exist $n, \tilde{x}$, a (possibly empty) sequence $\tilde{d}$ such that $c \notin \tilde{d}$, a process P and a (possibly empty) network M_1 such that: $M \equiv (\nu\tilde{d})(n[\mathtt{in}(c, \tilde{x})P]_l \mid M_1)$ and $M' \equiv (\nu\tilde{d})(n[c(\tilde{x}).P]_l | M_1)$.*
2. *If $M \xrightarrow{c?\vartheta@l} [\![M']\!]_\Delta$, then there exist $n, \tilde{x}$, a (possibly empty) sequence $\tilde{d}$ such that $c \notin \tilde{d}$, a process P and a (possibly empty) network M_1 such that $M \equiv (\nu\tilde{d})(n[c(x).P]_l | M_1)$ and $M' \equiv (\nu\tilde{d})(n[P\{\vartheta/\tilde{x}\}]_l | M_1)$.*
3. *If $M \xrightarrow{c_L![l,r]} [\![M']\!]_\Delta$, then there exist $n, \tilde{v}$, a (possibly empty) sequence $\tilde{d}$ such that $c \notin \tilde{d}$, a process P, two (possibly empty) sets J and K such that $\forall h \in J \cup K\ d(l, l_h) \leq r$ and a (possibly empty) network M_1 such that: $M \equiv (\nu\tilde{d})(n[\mathtt{out}\langle c_{L,r}, \tilde{v}\rangle.P]_l | \prod_{j\in J} n_j[\mathtt{in}(c, \tilde{x}_j).P_j]_{l_j} | \prod_{k\in K} n_k[c(\tilde{x}_k).P_k]_{l_k} | M_1)$ and $M' \equiv (\nu\tilde{d})(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l | \prod_{j\in J} n_j[c(\tilde{x}_j).P_j]_{l_j} | \prod_{k\in K} n_K[P_k\{\perp/\tilde{x}_k\}]_{l_k} | M_1)$.*
4. *If $M \xrightarrow{c_L!\tilde{v}[l,r]} [\![M']\!]_\Delta$, then there exist n, a (possibly empty) sequence $\tilde{d}$ such that $c \notin \tilde{d}$, a process P, a (possibly empty) set J, such that $\forall j \in J\ d(l, l_j) \leq r$ and a (possibly empty) network M_1 such that: $M \equiv (\nu\tilde{d})(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l | \prod_{j\in J} n_j[c(\tilde{x}_j).P_j]_{l_j} \mid M_1)$ and $M' \equiv (\nu\tilde{d})(n[P]_l | \prod_{j\in J} n_j[P_j\{\tilde{v}/\tilde{x}_j\}]_{l_j} \mid M_1)$.*

Proof. See Appendix. □

Now we show that structural congruence respects the transitions of Table 5.

Lemma 3.2. *If $M \xrightarrow{\gamma} [\![M']\!]_\theta$ and $M \equiv N$, then there exists N' such that $N \xrightarrow{\gamma} [\![N']\!]_\theta$ and $M' \equiv N'$.*

Proof. By induction on the depth of the inference $M \xrightarrow{\gamma} [\![M']\!]_\theta$. □

The following theorem establishes the relationship between the reduction semantics and the LTS one.

Theorem 3.3 (*Harmony*)**.** *Let M be a network.*

1. *If $M \to [\![M']\!]_\theta$ then there exist N and N' such that $N \xrightarrow{\tau} [\![N']\!]_\theta$, $M \equiv N$ and $M' \equiv N'$.*
2. *$M\downarrow_{c@K}$ iff M is well-formed and $N \xrightarrow{c!\tilde{v}@K\lhd R} [\![M']\!]_\Delta$ for some $R, \tilde{v}, N \equiv M$ and M'.*
3. *If $M \xrightarrow{\tau} [\![M']\!]_\theta$ then $M \to [\![M']\!]_\theta$.*
4. *If $M \xrightarrow{c!\tilde{v}@K\lhd R} [\![M']\!]_\Delta$ then $M \to [\![M']\!]_\Delta$.*

Proof. See Appendix. □

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

3.2. Probabilistic labelled bisimilarity

As for the previous versions of the calculus, we define a probabilistic labelled bisimilarity that is a complete characterization of our *probabilistic observational congruence*. It is built upon the following actions:

$$\alpha ::= c?@l \mid c?\vartheta@l \mid c!\tilde{v}@K \lhd R \mid \tau.$$

Again, we write $M \xrightarrow{\alpha}_\theta N$ if $M \xrightarrow{\alpha} [\![M']\!]_\theta$ and N is in the support of $[\![M']\!]_\theta$. Moreover we write $M \xrightarrow{\alpha} N$ if $M \xrightarrow{\alpha}_\theta N$ for some θ. A labelled *execution* e of a network M is a finite (or infinite) sequence of steps: $M \xrightarrow{\alpha_1}_{\theta_1} M_1 \xrightarrow{\alpha_2}_{\theta_2} M_2 \cdots \xrightarrow{\alpha_k}_{\theta_k} M_k$. With abuse of notation, we define $Exec_M$, $last(e)$, e^j and $e\uparrow$ as for unlabelled executions. We denote by $lbehave(M)$ the set of all possible behaviours of M, i.e., $lbehave(M) = \{(\alpha, [\![M']\!]_\theta) \mid M \xrightarrow{\alpha} [\![M']\!]_\theta\}$. Labelled executions arise by resolving the non-determinism of both α and $[\![M]\!]_\theta$. As a consequence, a scheduler[3] for the labelled semantics is a function F assigning a pair $(\alpha, [\![M]\!]_\theta) \in lbehave(last(e))$ with a finite labelled execution e. We denote by *LSched* the set of (admissible) schedulers for the LTS semantics, i.e., the set of all the schedulers F such that, for each network M in the support of F, M is well formed. Given a network M and a scheduler $F \in LSched$, we define $Exec^F_M$ as the set of all labelled executions starting from M and driven by F.

Since we are interested in weak observational equivalences, that abstract over τ-actions, we introduce the notion of *weak action*.

Definition 3.4 (*Weak Action*)**.** We denote by $\Longrightarrow$ the transitive and reflexive closure of $\xrightarrow{\tau}$ and by $\overset{\alpha}{\Longrightarrow}$ the weak action $\Longrightarrow\xrightarrow{\alpha}\Longrightarrow$. We denote by $\overset{\hat{\alpha}}{\Longrightarrow}$ the weak action $\overset{\alpha}{\Longrightarrow}$ if $\alpha \neq \tau$, and $\Longrightarrow$ otherwise.

In the following we will give the definition of probabilistic labelled bisimilarity with respect to a given set of schedulers.

Definition 3.5. Given a network M_0 and an admissible scheduler $F \in Sched$, we denote by $\hat{F}^{M_0}_{\mathcal{C}} \subseteq LSched$ the set of admissible schedulers $\hat{F} \in LSched$ such that $\forall e \in Exec^{\hat{F}}_{M_0}$ of the form

$$e = M_0 \xrightarrow{\alpha_1}_{\theta_1} M_1 \cdots \xrightarrow{\alpha_h}_{\theta_h} M_h$$

$\exists F' \in F^{M_0}_{\mathcal{C}}$, a context C_0 and $e' \in Exec^{F'}_{C_0[O_0]}$ with $O_0 \equiv M_0$ such that

$$e' = C_0[O_0] \rightarrow_{\theta'_1} C_1[O_1] \cdots \rightarrow_{\theta'_k} C_k[O_k]$$

and there exists a monotone surjective function f from $[0..k]$ to $[0..h]$ such that:

(i) $\forall i \in [0..k]$, $O_i \equiv M_{f(i)}$

(ii) $\forall j \in [1..k]$, $\theta'_j = \theta_{f(j)}$ when $M_{f(j-1)} \xrightarrow{\alpha_{f(j)}}_{\theta_{f(j)}} M_{f(j)}$.

Given a set $\mathcal{F} \subseteq Sched$ of schedulers and a network M_0, we define $\hat{\mathcal{F}}^{M_0}_{\mathcal{C}} = \bigcup_{F\in\mathcal{F}} \hat{F}^{M_0}_{\mathcal{C}}$.

Example 3.6. Consider the networks M_0 and N_0, and the schedulers F and F_1 introduced in the Example 2.8. If we take $\hat{F}_1 \in LSched$ such that

$$M_0 \xrightarrow{c_L![l,r]}_\Delta M_1 \xrightarrow{c_L!v[l,r]}_\Delta M_2 \in Exec^{\hat{F}_1^{M_0}}_{M_0},$$

then, since

$$M_0 \rightarrow_\Delta M_1 \rightarrow_\Delta M_2 \in Exec^F_{M_0}$$

the conditions of Definition 3.5 are satisfied when considering the empty context $C[\cdot] = \mathbf{0} \mid \cdot$ and the identity function $f(i) = i$ for $i \in [0..2]$. Hence $\hat{F}_1$ is a candidate for being in $\hat{F}^{M_0}_{\mathcal{C}}$.

Moreover, if we consider $\hat{F}_2 \in LSched$ such that

$$N_0 \xrightarrow{c?@k}_\Delta N_1 \xrightarrow{c?v@k}_\Delta N_2 \in Exec^{\hat{F}_2}_{N_0},$$

since

$$M_0 \mid N_0 \rightarrow_\Delta M_1 \mid N_1 \rightarrow_\Delta M_2 \mid N_2 \in Exec^{F_1}_{M_0|N_0}$$

with $F_1 \in F^{M_0}_{\mathcal{C}}$, by considering the contexts $C_i[\cdot] \equiv M_i \mid \cdot$ for $i \in [0..2]$, and the identity function $f(i) = i$ for $i \in [0..2]$ we get that $\hat{F}_2$ is a candidate for being in $\hat{F}^{M_0}_{\mathcal{C}}$.

[3] We abuse notation and still use F to denote a scheduler for the LTS semantics.

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

Proposition 3.7. 1. $Sched_{\mathcal{C}} = Sched$
2. $\widehat{Sched}_{\mathcal{C}} = LSched$

Proof. 1. The Proof follows straightforwardly from Definition 2.7.
2. $\forall F \in LSched, \forall M_0 \in \mathcal{N}$ and $\forall e \in Exec^F_{M_0}$ of the form:

$$e = M_0 \xrightarrow{\alpha_1}_{\theta_1} M_1 \cdots \xrightarrow{\alpha_k}_{\theta_k} M_k$$

it is always possible to find a context $C_0[\cdot]$ and a scheduler $F' \in LSched$ such that $e' \in Exec^{F'}_{C_0[M_0]}$ with

$$e' = C_0[M_0] \xrightarrow{\tau}_{\theta_1} \cdots C_1[M_1] \cdots \xrightarrow{\tau}_{\theta_k} C_k[M_k].$$

By Theorem 3.3, $\exists F'' \in Sched$ such that $e'' \in Exec^{F''}_{C_0[M_0]}$ with

$$e'' = C_0[M_0] \rightarrow_{\theta_1} \cdots C_1[M_1] \cdots \rightarrow C_k[M_k],$$

meaning that $F \in \widehat{Sched}_{\mathcal{C}}$ as required. □

In the probabilistic setting, while considering a computation with observable content, it is necessary to take into account the actual probability of this computation to ensure that weakly bisimilar systems may not only match one another's transitions but also perform these transitions with matching probabilities. To achieve this, we denote by $Exec^F_M(\stackrel{\alpha}{\Longrightarrow}, H)$ the set of executions that, starting from M, according to the scheduler F, lead to a network in the set H by performing $\stackrel{\alpha}{\Longrightarrow}$. Moreover, we define the probability of reaching a network in H from M by performing $\stackrel{\alpha}{\Longrightarrow}$, according to a scheduler F as $Prob^F_M(\stackrel{\alpha}{\Longrightarrow}, H) = Prob^F_M(Exec^F_M(\stackrel{\alpha}{\Longrightarrow}, H))$.

Definition 3.8 (*Probabilistic Labelled Bisimilarity*)**.** Let M and N be two networks. An equivalence relation $\mathcal{R}$ over networks is a *probabilistic labelled bisimulation* w.r.t. $\mathcal{F}$ if $M \mathcal{R} N$ implies: for all scheduler $F \in \hat{\mathcal{F}}^M_{\mathcal{C}}$ there exists a scheduler $F' \in \hat{\mathcal{F}}^N_{\mathcal{C}}$ such that for all α and for all classes $\mathcal{C}$ in $\mathcal{N}/\mathcal{R}$ it holds:

1. if $\alpha = \tau$ or $\alpha = c!\tilde{v}@K \lhd R$ then $Prob^F_M(\stackrel{\alpha}{\rightarrow}, \mathcal{C}) = Prob^{F'}_N(\stackrel{\hat{\alpha}}{\Longrightarrow} \mathcal{C})$;
2. if $\alpha = c?@l$ or $\alpha = c?\vartheta@l$ then either $Prob^F_M(\stackrel{\alpha}{\rightarrow}, \mathcal{C}) = Prob^{F'}_N(\stackrel{\alpha}{\Longrightarrow}, \mathcal{C})$ or $Prob^F_M(\stackrel{\alpha}{\rightarrow}, \mathcal{C}) = Prob^{F'}_N(\Longrightarrow, \mathcal{C})$.

Probabilistic labelled bisimilarity, written $\approx^{\mathcal{F}}_p$, is the largest probabilistic labelled bisimulation w.r.t. $\mathcal{F}$ over networks.

Notice that, in the above definition, input actions are allowed to be matched by τ actions. This reflects the fact that reception of messages cannot be directly observed by an external observer (see, e.g., [6]).

We prove that our probabilistic labelled bisimulation is a complete characterization of our notion of probabilistic barbed congruence.

The following proposition will be useful.

Proposition 3.9. *Let M and N be two networks. If $M \mathcal{R} N$ for some bisimulation $\mathcal{R}$ w.r.t. $\mathcal{F}$, then for all schedulers $F \in \hat{\mathcal{F}}^M_{\mathcal{C}}$ there exists a scheduler $F' \in \hat{\mathcal{F}}^N_{\mathcal{C}}$ such that for all α and for all classes $\mathcal{C}$ in $\mathcal{N}/\mathcal{R}$ it holds:*

1. *if $\alpha = \tau$ or $\alpha = c!\tilde{v}@K \lhd R$ then $Prob^F_M(\stackrel{\hat{\alpha}}{\Longrightarrow}, \mathcal{C}) = Prob^{F'}_N(\stackrel{\hat{\alpha}}{\Longrightarrow}, \mathcal{C})$;*
2. *if $\alpha = c?@l$ or $\alpha = c?\vartheta@l$ then either $Prob^F_M(\stackrel{\alpha}{\Longrightarrow}, \mathcal{C}) = Prob^{F'}_N(\stackrel{\alpha}{\Longrightarrow}, \mathcal{C})$ or $Prob^F_M(\stackrel{\alpha}{\Longrightarrow}, \mathcal{C}) = Prob^{F'}_N(\Longrightarrow, \mathcal{C})$.*

Proof. The proof follows by induction on the length of the weak transition $\stackrel{\hat{\alpha}}{\Longrightarrow}$. □

We can now prove that our bisimilarity is a proof method for our observational congruence, i.e., that $\approx^{\mathcal{F}}_p$ is contained in $\cong^{\mathcal{F}}_p$.

Theorem 3.10 (*Soundness*)**.** *Let M and N be two networks and $\mathcal{F} \subseteq Sched$. If $M \approx^{\mathcal{F}}_p N$ then $M \cong^{\mathcal{F}}_p N$.*

Proof. See Appendix. □

Finally, we prove that the observational congruence is contained in the labelled bisimilarity.

Theorem 3.11 (*Completeness*)**.** *Let M and N be two networks and $\mathcal{F} \subseteq Sched$. If $M \cong^{\mathcal{F}}_p N$ then $M \approx^{\mathcal{F}}_p N$.*

The following result is a consequence of Theorems 3.10 and 3.11.

Theorem 3.12 (*Characterization*)**.** *For every set $\mathcal{F} \subseteq Sched$, $\cong^{\mathcal{F}}_p = \approx^{\mathcal{F}}_p$.*

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

4. Interference metrics

We define a preorder over networks which allows us to compare the average level of interferences of networks exhibiting the same connectivity behaviour relative to a specific set of schedulers $\mathcal{F}$. We consider two metrics. The first focuses on emitters, and counts how many currently broadcasting nodes might interfere with each other due to an overlap in their communication ranges. The second metric is centred on receiver nodes and counts the number of active receivers which are simultaneously reached by two (or more) transmissions.

4.1. Sender-based interference

Let M be a network. Given a channel c, we denote by $\mathbf{Overlap^s}(M, c)$ the set of nodes currently broadcasting over c and whose transmission areas are overlapping at some locations. Formally, let

$$\mathtt{A}(M) \equiv (\nu \tilde{d})\left(\prod_{i\in I} n_i[\bar{c}_{L_i,r_i}\langle\tilde{v}\rangle.P_i]_{l_i} \mid \prod_{j\in J} n_j[c(\tilde{x}_j).P_j]_{l_j} \mid M'\right)$$

be the active nodes of M, where $c \notin \mathtt{Top}(M')$, then

$$\mathbf{Overlap^s}(M, c) = \{n_i \mid i \in I, \exists i' \in I.i \neq i' \text{ and } d(l_i, l_{i'}) \leq r_i + r_{i'}\}.$$

For example, consider the following network

$$\hat{M} = n_1[\mathtt{out}\langle c_{L_1,r_1}, \tilde{v}_1\rangle.P_1]_{l_1} \mid n_2[\bar{c}_{L_2,r_2}\langle\tilde{v}_2\rangle.P_2]_{l_2} \mid n_3[\bar{c}_{L_3,r_3}\langle\tilde{v}_3\rangle.P_3]_{l_3} \mid n_4[\bar{d}_{L,r}\langle\tilde{v}\rangle.P_4]_{l_4} \mid n_5[c(\tilde{x}).P_5]_{l_5} \mid n_6[\mathtt{in}(c, \tilde{y}).P_6]_{l_6}$$

where $d(l_i, l_{i'}) > r_i$ for all $i, i' \in \{1, 2, 3\}$ with $i \neq i'$, i.e., the nodes n_1, n_2, and n_3 are all far enough away from each other and can broadcast at the same time over the channel c. In this case, function $\mathbf{Overlap^s}(\hat{M}, c)$ is defined as follows: for all $c' \neq c$ (e.g., $c' = d$) $\mathbf{Overlap^s}(\hat{M}, c') = \emptyset$, while

$$\mathbf{Overlap^s}(\hat{M}, c) = \begin{cases} \{n_2, n_3\} & \text{if } d(l_2, l_3) \leq r_2 + r_3 \\ \emptyset & \text{otherwise.} \end{cases}$$

We define the sender-based level of interference induced by a probabilistic *transition* as follows:

$$\mathbf{Interf^s}(M, N) = \begin{cases} |\mathbf{Overlap^s}(N, c)| - |\mathbf{Overlap^s}(M, c)| \\ \quad \text{if } M \xrightarrow{c_L![l,r]} [\![N]\!]_\Delta \quad \text{for some } L, l, r; \\ 0 \quad \text{otherwise.} \end{cases}$$

Consider again the above network $\hat{M}$. Since $d(l_1, l_i) > r_1$ for $i \in \{2, 3\}$, we have $\hat{M} \xrightarrow{c_{L_1}![l_1,r_1]} [\![\hat{N}]\!]_\Delta$, where

$$\hat{N} = n_1[\bar{c}_{L_1,r_1}\langle\tilde{v}_1\rangle.P_1]_{l_1} \mid n_2[\bar{c}_{L_2,r_2}\langle\tilde{v}_2\rangle.P_2]_{l_2} \mid n_3[\bar{c}_{L_3,r_3}\langle\tilde{v}_3\rangle.P_3]_{l_3} \mid n_4[\bar{d}_{L,r}\langle\tilde{v}\rangle.P_4]_{l_4} \mid n_5[P'_5]_{l_5} \mid n_6[P'_6]_{l_6}.$$

The sender-based level of interference induced by $\hat{M} \xrightarrow{c_{L_1}![l_1,r_1]} [\![\hat{N}]\!]_\Delta$ is, e.g.:

- If n_1 is too far away from both n_2 and n_3, i.e., $d(l_1, l_j) > r_1 + r_j$ for $j \in \{2, 3\}$, then $\mathbf{Overlap^s}(\hat{N}, c) = \mathbf{Overlap^s}(\hat{M}, c)$. Hence:
$$\mathbf{Interf^s}(\hat{M}, \hat{N}) = 0.$$
- If n_2 and n_3 were already overlapping, i.e., $d(l_2, l_3) \leq r_2 + r_3$ and n_1 is not too far away from at least one of them, i.e., $d(l_1, l_2) \leq r_1 + r_2$ or $d(l_1, l_3) \leq r_1 + r_3$ then $\mathbf{Overlap^s}(\hat{N}, c) = \{n_1, n_2, n_3\}$. Therefore, $\mathbf{Interf^s}(\hat{M}, \hat{N}) = 1$. The additional potentially disturbed communication is the one just started by n_1.
- If n_2 and n_3 were not overlapping, but n_1 is not too far away of both of them, then $\mathbf{Overlap^s}(\hat{N}, c) = \{n_1, n_2, n_3\}$. Thus, $\mathbf{Interf^s}(M, N) = 3$. Here the started broadcast by n_1 overlaps with both the *previously safe* existing transmission areas.
- Finally, n_2 and n_3 were not overlapping, but n_1 is not too far away of exactly one of them (e.g., n_2), then $\mathbf{Overlap^s}(\hat{N}, c) = \{n_1, n_2\}$, and $\mathbf{Interf^s}(M, N) = 2$.

4.2. Receiver-based interference

Hereafter, we denote by $\mathbf{Coll^r}(M, c, l, r)$ the set of nodes in M which are currently listening over channel c and lie in the transmission range of a sender located at l with radius r. Formally, let $\mathtt{A}(M) \equiv (\nu\tilde{d})\left(\prod_{i\in I} n_i[\bar{c}_{L_i,r_i}\langle\tilde{v}\rangle.P_i]_{l_i} \mid \prod_{j\in J} n_j[c(\tilde{x}_j).P_j]_{l_j} \mid M'\right)$ be the active nodes of M, where $c \notin \mathtt{Top}(M')$, then

$$\mathbf{Coll^r}(M, c, l, r) = \{n_j | j \in J \text{ and } d(l, l_j) \leq r\}.$$

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

The number of receiver-based interferences induced by a probabilistic step is defined as follows:

$$\mathbf{Interf}^{\mathbf{r}}(M,N) = \begin{cases} |\mathbf{Coll}^{\mathbf{r}}(M,c,l,r)| \\ \quad \text{if } M \xrightarrow{c_L?[l,r]} [\![N]\!]_{\Delta} \quad \text{for some } L; \\ 0 \quad \text{otherwise.} \end{cases}$$

For instance, if we consider again our previous networks $\hat{M}$ and $\hat{N}$, assuming that n_1 can reach both l_5 and l_6 then $P_5' = P_5\{\perp/\tilde{x}\}$ and $P_6' = c(\tilde{y}).P_6$. Then, $\mathbf{Coll}^{\mathbf{r}}(\hat{M}, c, l_1, r_1) = \{n_5\}$. Hence $\mathbf{Interf}^{\mathbf{r}}(\hat{M}, \hat{N}) = 1$.

Now, let $\chi \in \{\mathbf{s}, \mathbf{r}\}$. The χ-type number of interferences induced by an execution $e = M_0 \xrightarrow{\alpha_1}_{\theta_1} M_1 \cdots \xrightarrow{\alpha_k}_{\theta_k} M_k$ is

$$\mathbf{Interf}^{\chi}(e) = \sum_{i=1}^{k} \mathbf{Interf}^{\chi}(M_{i-1}, M_i).$$

Let H be a set of networks, we denote by $Paths_M^F(H)$ the set of all executions from M ending in H and driven by F which are not prefixes of any other execution ending in H. Formally, $Paths_M^F(H) = \{e \in Exec_M^F(H) \mid last(e) \in H \text{ and } \forall e' \text{ such that } e \text{ is a prefix of } e',\ e' \notin Paths_M^F(H)\}$. The average number of interferences is computed by weighting the number of interferences of each execution by its probability according to F and normalized by the overall probability of reaching H.

Definition 4.1. Let H be a set of networks. The average number of interferences to reach H from M according to scheduler F is

$$\mathbf{Interf}^{\chi}_{M,F}(H) = \frac{\sum_{e \in Paths_M^F(H)} \mathbf{Interf}^{\chi}(e) \times P_M^F(e)}{\sum_{e \in Paths_M^F(H)} P_M^F(e)}$$

Definition 4.2. Let $\mathcal{H}$ be a countable set of sets of networks and $\mathcal{F}$ a set of schedulers. We say that N is *at least as interference efficient as* M relative to $\mathcal{H}$ and $\mathcal{F}$, written

$$N \sqsubseteq^{\chi}_{\langle \mathcal{H}, \mathcal{F} \rangle} M,$$

if $N \approx_p^{\mathcal{F}} M$ and, for all $H \in \mathcal{H}$ and for all schedulers $F \in \mathcal{F}$, there exists a scheduler $F' \in \mathcal{F}$ such that $\mathbf{Interf}^{\chi}_{N,F'}(H) \leq \mathbf{Interf}^{\chi}_{M,F}(H)$.

5. Case study: the alternating bit protocol

The alternating bit protocol (ABP) is a simple network protocol designed to achieve a point to point reliable transmission on unreliable channels. Messages are sent from a transmitter to a receiver and include the payload (i.e., the meaningful data) and some control information (e.g., the address identifying the destination, a checksum for the integrity checks, etc.). Among the control information, there is packet sequence number of 1 bit. When the sender sends a message with sequence number b, it waits for an acknowledge (`ack`) identified with the same sequence number from the receiver. If the `ack` does not arrive before a given deadline then the sender assumes that the packet has been lost and tries to resent it. The deadline is chosen according to the channel characteristics and must be greater than its round trip time. When the `ack` is received correctly, the sender flips the sequence number and starts a new transmission.

We consider a network consisting of two mobile sender nodes, n_1 and n_2, communicating with a static receiver node m. Node n_1 moves back and forth between locations l_1 and l_2 according to the probability distribution defined by the discrete time homogeneous Markov chain with the following transition matrix (where $0 < p, q < 1$):

$$\mathbf{J} = \begin{vmatrix} 1-p & p \\ q & 1-q. \end{vmatrix}$$

Node n_2 moves similarly between l_3 and l_4 according to a discrete time Markov chain with the same transition matrix $\mathbf{J}$. We also assume that the receiver node is always in the transmission range of both senders (and that the senders are always in the range of the receiver) regardless of where the senders are located. This guarantees that m receives any packet from the senders (unless a collision occurs), and that both senders receive any `ack` sent by m.

Furthermore, we assume that the transmission ranges of the senders overlap only when n_1 is at l_1 and n_2 is at l_3. As a result, unless n_1 is at l_1 and n_2 is at l_3, the senders are in the condition to attempt a simultaneous transmission (as they do not sense each other) leading to an interference (see Fig. 1): in literature, this is known as the *hidden station problem*. Notice that while communications can be damaged by many factors, we shall consider only the interference factor in this analysis.

Table 6 shows an encoding of the sender and receiver processes. SND_j runs inside node n_j, sending a queue of messages T_j with sequence bit b_j; RCV, in turn, runs inside the receiver node m, expecting messages with sequence bits b_1 and b_2 from

Table 6
ABP.

$SND_j\langle b_j, T_j\rangle =$	$\texttt{[empty}(T) = \texttt{false]}(\texttt{out}\langle c_{\{k\},r_j}, (b_j, \texttt{head}(T_j), n_j)\rangle.WAIT\_Ack_j\langle b_j, T_j\rangle),$ $\texttt{out}\langle \texttt{ok}_{\{k\},r_j}, (n_j, END)\rangle$
$WAIT\_Ack_j\langle b_j, T_j\rangle =$	$\texttt{in}(c, (x, y, z)).[y = n_j]([(x = b_j) \wedge (z = ACK)]SND_j\langle \neg b_j, \texttt{dequeue}(T_j)\rangle),$ $SND_j\langle b_j, T_j\rangle), WAIT\_Ack_j\langle b_j, T_j\rangle$
$RCV\langle b_1, b_2\rangle =$	$\texttt{in}(c, (x, y, z)).[z = n_1]\big(([x = b_1]\texttt{out}\langle c_{\{l_1,l_2,l_3,l_4\},r}, (b_1, n_1, ACK)\rangle.$ $RCV\langle \neg b_1, b_2\rangle, \texttt{out}\langle c_{\{l_1,l_2,l_3,l_4\},r}, (b_1, n_1, NACK)\rangle.RCV\langle b_1, b_2\rangle),$ $[z = n_2]([x = b_2]\texttt{out}\langle c_{\{l_1,l_2,l_3,l_4\},r}, (b_2, n_2, ACK)\rangle.RCV\langle b_1, \neg b_2\rangle,$ $\texttt{out}\langle c_{\{l_1,l_2,l_3,l_4\},r}, (b_2, n_2, NACK)\rangle.RCV\langle b_1, b_2\rangle)\big),$ $\texttt{out}\langle c_{\{l_1,l_2,l_3,l_4\},r}, (b_1, n_1, NACK)\rangle.\texttt{out}\langle c_{\{l_1,l_2,l_3,l_4\},r}, (b_2, n_2, NACK)\rangle.$ $RCV\langle b_1, b_2\rangle$
$ABP =$	$(\nu c)\big(n_1[SND_1\langle 1, T_1\rangle]_{l_1} \mid n_2[SND_2\langle 1, T_2\rangle]_{l_3}\ m[RCV\langle 1, 1\rangle]_k\big)$

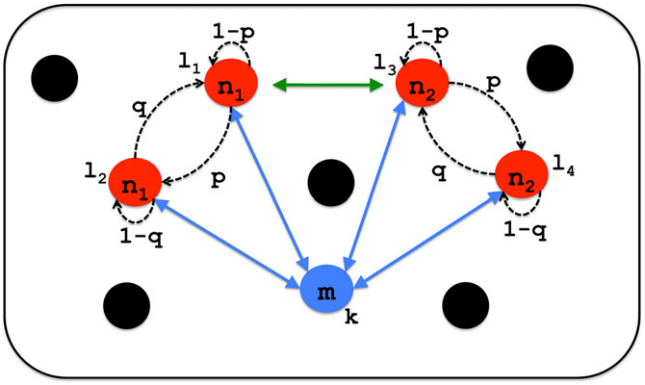


Fig. 1. Graphical representation of node mobility.

Table 7
SIC_ABP.

$RCV_{SIC}\langle b_1, b_2\rangle =$	$\texttt{in}(c, (x_1, x_2, x_3))[x_3 = n_1]\big(([x_1 = b_1]\texttt{out}\langle c_{\{l_1,l_2,l_3,l_4\},r}, (b_1, n_1, ACK)\rangle.$ $RCV_{SIC}\langle \neg b_1, b_2\rangle, \texttt{out}\langle c_{\{l_1,l_2,l_3,l_4\},r}, (b_1, n_1, NACK)\rangle.RCV_{SIC}\langle b_1, b_2\rangle),$ $[x_3 = n_2]([x_1 = b_2]\texttt{out}\langle c_{\{l_1,l_2,l_3,l_4\},r}, (b_2, n_2, ACK)\rangle.RCV_{SIC}\langle b_1, \neg b_2\rangle,$ $\texttt{out}\langle c_{\{l_1,l_2,l_3,l_4\},r}, (b_2, n_2, NACK)\rangle.RCV_{SIC}\langle b_1, b_2\rangle)\big),$ $\texttt{out}\langle c_{\{l_1,l_2,l_3,l_4\},r}, (b_1, n_1, NACK)\rangle.WAIT\langle \perp_{x_1,x_2,x_3}, b_1, b_2\rangle$
$WAIT\langle \perp_{p_1,p_2,p_3}, b_1, b_2\rangle =$	$\texttt{in}(c, (x_1, x_2, x_3))[x_3 = n_1][x_1 = b_1](\texttt{out}\langle c_{\{l_1,l_2,l_3,l_4\},r}, (b_1, n_1, ACK)\rangle.$ $[f(x_3, p_3) = n_2][b_2 = f(x_1, p_1)](\texttt{out}\langle c_{\{l_1,l_2,l_3,l_4\},r}, (b_2, n_2, ACK)\rangle.$ $RCV_{SIC}\langle \neg b_1, \neg b_2\rangle), \texttt{out}\langle c_{\{l_1,l_2,l_3,l_4\},r}, (b_2, n_2, NACK)\rangle.RCV_{SIC}\langle \neg b_1, b_2\rangle),$ $\texttt{out}\langle c_{\{l_1,l_2,l_3,l_4\},r}, (x_1, n_1, NACK)\rangle.WAIT\langle \perp_{x_1,x_2,x_3}, b_1, b_2\rangle$
$SIC\_ABP =$	$(\nu c)\big(n_1[SND_1\langle 1, T_1\rangle]_{l_1} \mid n_2[SND_2\langle 1, T_2\rangle]_{l_3} \mid m[RCV_{SIC}\langle 1, 1\rangle]_k\big)$

n_1 and n_2, respectively. We presuppose few auxiliary functions: `empty()`, `dequeue()` and `head()` implement the standard queue operations, while $\neg b$ flips the value of the bit b. Finally, `ok` is a channel name and a location introduced for the purposes of our analysis.

5.1. Successive interference cancellation (SIC) for CDMA

Here, we sketch a simplified version of the successive interference cancellation (SIC) method for CDMA/CA [2] transmission scheme. Assume that nodes n_1 and n_2 cause an interference at m by sending packets encoded by signals x_A and x_B. Node m receives the signal $y_1 = x_A + x_B$, detects the interference and stores y_1 in memory. In the successive time slot, n_1 successfully resends x_A, i.e., m receives $y_2 = x_A$ and sends an `ack` to n_1. Now, x_B may be extracted from y_1 by m without further retransmissions as the result of $y_1 - x_A$. Although in practice this procedure is not always successful, we assume that messages can always be recovered correctly.

In modelling this protocol, the sender processes remain the same as in the simple *ABP* protocol defined in Table 6, while the receiver process is defined as shown in Table 7.

In order to compare the observational behaviours of the protocols, we assume that a successful completion of transmission of the packets by a sender, indicated by broadcasting the message "END" over the channel `ok`, is observable to any observer node located at k. In this analysis, we are only interested in the levels of interference due to the internal nodes of the protocols. Therefore, we restrict communications over the channel c to the internal nodes of the protocols.

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

5.2. Measuring the interference level of the protocols

Schedulers constitute an essential feature for modelling communication protocols as they provide freedom in modelling implementation and incomplete knowledge of the system. However, many schedulers could be unrealistic or useless. Indeed, schedulers giving priority to communications over movements will, for instance, cancel the two-state nature of the sender nodes, while those giving priority to end broadcasting actions over begin broadcasting actions will prevent any interference. Therefore, we consider the following set $\mathcal{F}_{\mathbf{fas}}$ of *fair alternating schedulers* which:

1. always alternate between sending packets and node movements so that at each interaction of the transmitters with the receiver, the formers could be far enough away from each other to cause interference or not;
2. give priority to acknowledgement actions (*ACK* and *NACK*) to model our assumption of an error-free feedback channel;
3. give priority to begin broadcasting actions (Beg-Bcast) over end broadcasting actions (End-Bcast).

Notice that the analysis of the model under the set of fair alternating schedulers is general because it establishes a relative speed between the packet transmissions and node movements that, in practice, can be regulated by means of the transition probabilities of **J**. Moreover, all the events that may influence the performance of the protocols, and in particular the interferences, are allowed.

We now prove some preliminary results needed to show that applying the SIC method to the alternating bit protocol reduces the level of interference in the system. We first prove that the two networks exhibit the same observable behaviour relative to $\mathcal{F}_{\mathbf{fas}}$.

Proposition 5.1. *ABP* $\approx_p^{\mathcal{F}_{\mathbf{fas}}}$ *SIC_ABP*.

Proof. For brevity, we give just a sketch of the proof. In both protocols, the only observable actions, are the final messages sent by n_1 and n_2 through the channel ok, that occur when all the messages of their respective queues are completely and correctly received by m, since the other actions are either silent, or hidden by the restriction operator applied to the channel c. Hence, in both protocols the only observable actions are of the form:

$$\Longrightarrow \xrightarrow{\mathtt{ok}!(n_1,END)@k\lhd k} \Longrightarrow,$$

or

$$\Longrightarrow \xrightarrow{\mathtt{ok}!(n_2,END)@k\lhd k} \Longrightarrow .$$

We can conclude that *ABP* and *SIC_ABP* are probabilistic bisimilar, because they exhibit the same behaviour, with the same probability. Indeed, the characteristics of matrix **J** ensures that for both the protocols the probability of eventually transmitting the whole queue of messages in 1. □

Now let T_1 and T_2 be the queues of messages to be transmitted by the senders. We compare the interference efficiency of the protocols in the context of the set $\mathcal{H}(T_1, T_2) = \{H_\rho(T_1, T_2) \mid \rho \leq \max(|T_1|, |T_2|)\}$ where $H_\rho(T_1, T_2)$ means that all the packets up to ρ have been correctly transmitted by both senders and is defined as $H_\rho(T_1, T_2) = H^1_\rho(T_1, T_2) \cup H^2_\rho(T_1, T_2)$ where

$$H^1_\rho(T_1, T_2) = \left\{M \mid M \equiv (\nu c)\left(n_1[SND_1\langle b_1, \mathtt{dequeue}^\rho(T_1)\rangle]_{l'} \mid n_2[SND_2\langle b_2, \mathtt{dequeue}^\rho(T_2)\rangle]_{k'} \mid m[RCV\langle b_1, b_2\rangle]_k\right)\right\}$$

with the assumption that $\mathtt{dequeue}(\emptyset) = \emptyset$, and $b_1,\ b_2 \in \{0, 1\}$. Similarly

$$H^2_\rho(T_1, T_2) = \left\{N \mid N \equiv (\nu c)\left(n_1[SND_1\langle b_1, \mathtt{dequeue}^\rho(T_2)\rangle]_{l''} \mid n_2[SND_2\langle b_2, \mathtt{dequeue}^\rho(T_2)\rangle]_{k''} \mid m[RCV_{SIC}\langle b_1, b_2\rangle]_k\right)\right\}$$

with b_1 and b_2 in $\{0, 1\}$, l', l'' in $\{l_1, l_2\}$, and k', k'' in $\{l_3, l_4\}$. Then, we compute the interference level of the protocols assuming that we start by a move action for each sender node so that their first transmissions could create an interference if they move too far away from each other.[4] The results are summarized in the following propositions.

Proposition 5.2. *For all F in* $\mathcal{F}_{\mathbf{fas}}$ *and for all* $\rho \leq \max(|T_1|, |T_2|)$ *we have:*

$$\mathbf{Interf}^{\mathbf{s}}_{ABP,F}(H_\rho(T_1, T_2)) = 2 \times \mathbf{Interf}^{\mathbf{r}}_{ABP,F}(H_\rho(T_1, T_2)) = 2 \times \left(\frac{(p+q)^2}{q^2} - 1\right) \times \min(\rho, |T_1|, |T_2|)$$

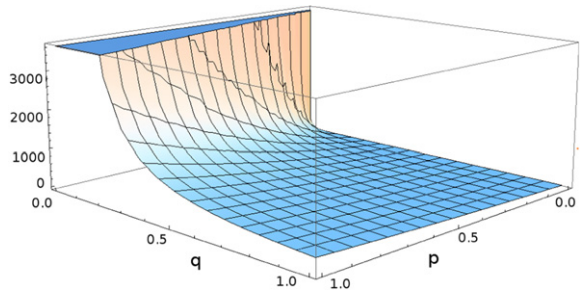
with $0 < p, q < 1$.

The proof relies on the observation that correct packets are sent only when the mobile nodes are in the locations l_1 and l_3. Hence, by exploiting the independence between the stochastic processes underlying the node movements, the result follows by standard analysis of absorbing Markov chains.

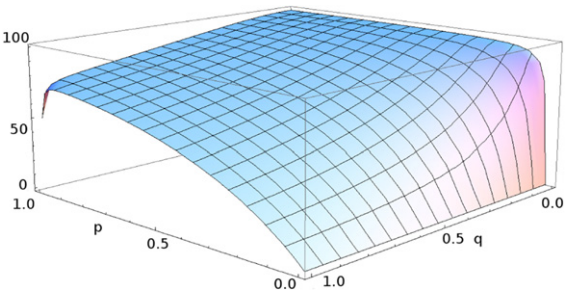
Note that our sender-based interference metric coincides with the number of lost packets. For the *ABP* with *SIC*, we have:

4 The analysis for the other case is similar.

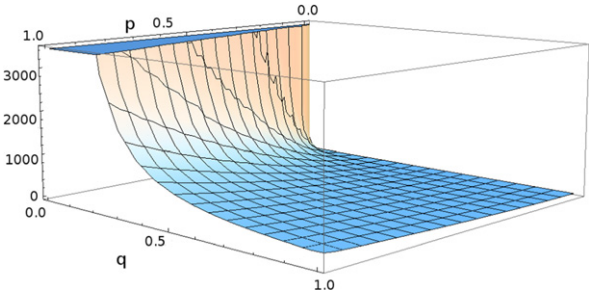
Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003



(a) Plotting of $\mathbf{Interf}^{\mathbf{r}}_{ABP,F}(H_\rho)$ given by Proposition 5.2.



(b) Plotting of $\mathbf{Interf}^{\mathbf{r}}_{SIC\_ABP,F}(H_\rho)$ given by Proposition 5.3.



(c) Plotting of $\mathbf{Interf}^{\mathbf{r}}_{ABP,F}(H_\rho) - \mathbf{Interf}^{\mathbf{r}}_{SIC\_ABP,F}(H_\rho)$.

Fig. 2. Interference levels for *ABP* and *SIC_ABP* and their comparison.

Proposition 5.3. *For all F in* $\mathcal{F}_{\mathbf{fas}}$ *and each* $\rho \leq \max(|T_1|, |T_2|)$ *we have:*

$$\begin{aligned}\mathbf{Interf}^{\mathbf{s}}_{SIC\_ABP,F}(H_\rho(T_1,T_2)) &= 2 \times \mathbf{Interf}^{\mathbf{r}}_{SIC\_ABP,F}(H_\rho(T_1,T_2)) \\ &= 2 \times \frac{p}{(p+q)^3}\left(n(p+q)(p+2q)\right. \\ &\quad \left. - \frac{((1-p-q)^n-1)(p+q-1)(p^2-p(1-p-q)^{n+1}-4q+3pq+2q^2-p)}{p+q-2}\right) \\ &\quad \times \min(\rho, |T_1|, |T_2|)\end{aligned}$$

with $0 < p, q < 1$.

Also in this case the proof is based on standard transient Markov chain analysis and exploits the independence among the processes that regulate the node movements. Indeed, the nth steps transition probability matrix $(\mathbf{J})^n$ is:

$$(\mathbf{J})^n = \begin{vmatrix} \frac{p(1-p-q)^n+q}{p+q} & \frac{p-p(1-p-q)^n}{p+q} \\ \frac{q-q(1-p-q)^n}{p+q} & \frac{p+q(1-p-q)^n}{p+q} \end{vmatrix}.$$

According to the SIC specification, nodes need only to send one packet for a successful packet transmission if they are in the locations l_1 and l_3. All the other location combinations require one of the nodes to send two packets for each successful transmission (while the other sends just one). Starting from states l_1 and l_3, the probability of being still in the same state after $i > 0$ steps is given by $(p(1-p-q)^i+q)^2/(p+q)^2$ (by independence). We derive the expression given by Proposition 5.3 as the closed expression of the following sum which represents the expected number of observed interferences for sending n packets:

$$\sum_{i=1}^{n}\left(1-\left(\frac{p(1-p-q)^i+q}{p+q}\right)^2\right).$$

Let us denote by H_ρ the set $H_\rho(T_1, T_2)$. In Fig. 2 (a) and (b) we show a plot of $\mathbf{Interf}^{\mathbf{s}}_{ABP,F}(H_\rho)$ and $\mathbf{Interf}^{\mathbf{r}}_{SIC\_ABP,F}(H_\rho)$, respectively, as a function of p and q, for $\min(\rho, |T_1|, |T_2|) = 100$, while Fig. 2 (c) shows a plot of $\mathbf{Interf}^{\mathbf{r}}_{ABP,F}(H_\rho) - \mathbf{Interf}^{\mathbf{r}}_{SIC\_ABP,F}(H_\rho)$. Finally, from Propositions 5.1–5.3, we can conclude that the *SIC*-based *ABP* protocol is at least efficient as the flooding version in terms of interference.

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

Theorem 5.4. *SIC_ABP* $\sqsubseteq^{\chi}_{\langle \mathcal{F}_{\mathbf{fas}}, \mathcal{H}(T_1,T_2)\rangle}$ *ABP*.

Proof. Apply Propositions 5.1–5.3. □

6. Case study: location aided routing protocols

Our second case study shows how to exploit our framework to model a location based routing protocol, specifically the Location Aided Routing (LAR) [10]. Informally, location based routing algorithms assume that each node of the wireless network is aware of its own location thanks to a Global Positioning System (GPS) device or thanks to other mechanisms such as the knowledge of the distances between its location at a given epoch and some other static stations. The main idea behind the development of these algorithms is that in very large mobile networks using a flooding policy in an AODV style [9] may turn out to be very expensive in terms of number of sent packets and hence of energy consumption. Location based routing algorithms aim at controlling the flooding by guessing the possible location of the destination node. The guess can be driven by several factors, such as the knowledge of the destination node's location in the latest communication joint with some assumptions on the node's maximum movement speed. In this section, we show our framework at work on a simplified version of the LAR protocol, and prove that, under mild assumptions on the node mobility, it is equivalent to the flooding algorithm in terms of the probability of discovering a path. Obviously, it is not possible to establish a general interference preorder between the two protocols, but this can be done (algorithmically) for specific instances of wireless networks.

6.1. Simple flooding: description

Protocol LAR extends the route discovery based on flooding by exploiting information about locations within the network. The simplest route discovery algorithm based on flooding consists of three simple packets: request, reply and error [28], which are forwarded within the network. They are structured as follows:

- *Route Request* packet (RREQ) has the form:

  $(S, \mathit{Bid}, D, \texttt{seq\#}_S, \texttt{hop\_counter})$,

  where S is the permanent source address, *Bid* is the Request Id (unique identifier), D is the permanent address of the destination, $\texttt{seq\#}_S$ denotes the sequence number of the source, and `hop_counter` is the number of hops to reach the destination (which is initially set to 0 and then incremented at each request forwarding).
- *Route Reply* packet (RREP) has the form:

  $(S, \mathit{Bid}, D, \texttt{seq\#}_D, \texttt{hop\_counter}, \texttt{Lifetime})$,

  where S, *Bid* and D are as above, $\texttt{seq\#}_D$ is the sequence number of the destination, `hop_counter` is the number of hops to reach the destination and `Lifetime` is the duration of the route validity.
- *Route Error* packet (RERR) has the form:

  $(S, D, \texttt{seq\#}_D)$,

  where S D and $\texttt{seq\#}_D$ are as in the previous case.

Normally, a node looking for a path to a given destination, simply broadcasts a RREQ within the network. Having sent the packet, the node sets a timeout to manage the cases when the destination does not receive the request, or the reply packet is lost. If the timeout expires, the node broadcasts a new request, using a different sequence number to avoid loops. When the destination finally receives the RREQ, it immediately sends back the corresponding RREP, using unicast communication, i.e., each intermediate node forwards the RREP using the information in its routing table. When, during a communication, a node realizes that a link failed, it broadcasts a RERR and each node will update its routing table.

6.2. Exploiting location data: the LAR policy

LAR extends the simple flooding algorithm described above by directing the propagation of the discovery packets to a particular network area based on the expected locations of the destination node. In the LAR specification, the *Expected Zone* is the network area where the source expects to find the destination node. This is determined by means of the information that the source has previously retrieved about the destination location. In practice, if node S knows that destination node D was located at location l_1 at epoch t, and it moves with a speed v, then it can calculate the circle area centred at l_1, with radius $v(t' - t)$, where t' is the current epoch. If S does not know anything about D, then the *Expected Zone* coincides with the entire network.

The *Request Zone* is the network area that the source defines to specify a candidate route to the destination. An intermediate node forwards a route request only if it is within the *Request Zone*. There are different ways to define a *Request Zone*: usually choosing a smaller area reduces the message overhead (because it reduces the number of forwarded packets), while a larger area reduces the latency of the route discovery because the network finds a path with higher probability.

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

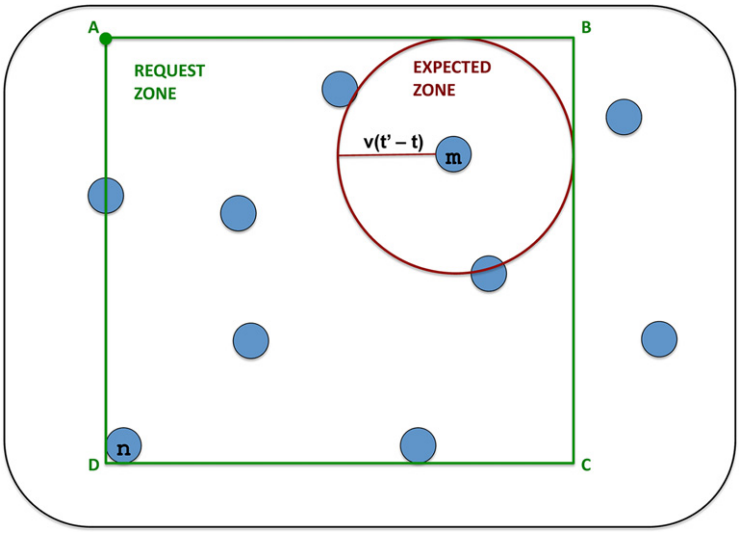


Fig. 3. *Expected* and *Request Zones* in the LAR protocol.

(a) A simple *route request* packet.

(b) A *route request* packet with location information.

Fig. 4. Different *route request* packets of LAR - *Scheme 1*.

LAR behaves similarly to the simple flooding, with the difference that a node that is not inside the *Request Zone* does not forward the request. LAR can use two different policies for determining the *Request Zone*: we focus on the first such policy, known as *LAR Scheme 1*.

LAR Scheme 1 uses a rectangular *Request Zone*, depending on the position of the source with respect to the *Expected Zone*. In particular, the *Request Zone* will be the smallest rectangle containing both the *Expected Zone* and the position of the source node, as shown in Fig. 3.

Let (X_S, Y_S) and (X_D, Y_D) the Cartesian coordinates of S and D, and R the radius of the *Expected Zone*. If S is outside the *Expected Zone*, the coordinates of the rectangle area are:

$$\text{A} :\rightarrow (X_S, Y_D + R) \qquad \text{B} :\rightarrow (X_D + R, Y_D + R)$$
$$\text{C} :\rightarrow (X_D + R, Y_S) \qquad \text{D} :\rightarrow (X_S, Y_S).$$

If S falls inside the *Expected Zone*, the coordinates of the rectangle area are:

$$\text{A} :\rightarrow (X_D - R, Y_D + R) \qquad \text{B} :\rightarrow (X_D + R, Y_D + R)$$
$$\text{C} :\rightarrow (X_D - R, Y_D - R) \qquad \text{D} :\rightarrow (X_D + R, Y_D - R).$$

When S broadcasts its request, it includes the coordinates of the *Request Zone* rectangle (see Fig. 4). Once an intermediate node receives a RREQ, this is discarded if its location does not fall within the rectangle specified in the packet. To take into account the location measuring error, a positive value e is added to the radius of the *Expected Zone*, consequently enlarging also the *Request Zone*.

6.3. Modelling the network

We encode the simple flooding and the LAR protocols using PEBUM. We abstract out all details about how the *Expected Zone* and *Request Zone* are determined, by using pre-defined functions that are implemented according to the specifications of LAR Scheme 1.

We first introduce some auxiliary functions to simplify the protocol specification:

- `gps`: returns the actual geographical position of the node executing the process (by means, e.g., of GPS technology);
- `dist`(l): returns the distance from location l and the location of the node executing the process;
- `self`: returns the name (permanent address) of the node executing the process;
- `geq`(k, l) = `true` if $k \geq l$, `false` otherwise;
- `inside`(s, A) = `true` if $s \in A$, `false` otherwise;
- `unable`(n) = refreshes the route table, removing the existing path to n;

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

Table 8
Process specifications used in the case study of Section 6.

$Q\_X = \mathtt{in}(c, x_1, x_2, x_3, x_4, x_5, x_6, x_7).$
$[x_1 = \mathtt{rreq}](\mathtt{[control}(x_3) = \mathtt{false}](\mathtt{[}x_4 = \mathtt{self}]$
$\mathtt{out}\langle c_{\mathtt{next\_hop}_{x_2},r}, (\mathtt{rrep}, s, \mathit{Bid}, d, \mathtt{seq\#}_s, \mathtt{hop\_counter})\rangle.Q\_X, \mathit{RREQ\_X}\langle\tilde{x}\rangle), Q\_X),$
$[x_1 = \mathtt{rrep}](\mathtt{[}x_2 = \mathtt{self}]\mathtt{out}\langle ud_{\mathtt{gps},r}, x_2, x_3, x_4, x_5, x_6, x_7\rangle,$
$\mathtt{out}\langle c_{\mathtt{next\_hop}_{x_2},r}, (\mathtt{rrep}, s, \mathit{Bid}, d, \mathtt{seq\#}_s, \mathtt{hop\_counter})\rangle.Q\_X),$
$[x_1 = \mathtt{rerr}]\mathtt{unable}(x_4).Q\_X, Q\_X$

$\mathit{RREQ\_SIMPLE}\langle(\mathtt{rreq}, s, \mathit{Bid}, d, \mathtt{seq\#}_s, \mathtt{hop\_counter})\rangle =$
$\quad[\mathtt{find\_path}(d) = \mathtt{true}].$
$\quad\mathtt{out}\langle c_{\mathtt{next\_hop}_d,r}, (\mathtt{rrep}, s, \mathit{Bid}, d, \mathtt{seq\#}_d, \mathtt{hop\_counter} + 1 + \mathtt{hopcount}_d, \mathtt{timeout})\rangle,$
$\quad\mathtt{out}\langle c_{\mathtt{Request\_Zone},r}, (\mathtt{rreq}, s, \mathit{Bid}, d, \mathtt{seq\#}_s, (\mathtt{hop\_counter}) + 1)\rangle.Q\_SIMPLE$

$\mathit{RREQ\_LAR1}\langle(\mathtt{rreq}, s, \mathit{Bid}, d, \mathtt{Request\_Zone}, \mathtt{seq\#}_s, \mathtt{hop\_counter})\rangle =$
$\quad([\mathtt{inside}(\mathtt{gps}, \mathtt{Request\_Zone}) = \mathit{true}]$
$([\mathtt{find\_path}(d) = \mathtt{true}]$
$\mathtt{out}\langle c_{\mathtt{next\_hop}_d,r}, (\mathtt{rrep}, s, \mathit{Bid}, d, \mathtt{seq\#}_d, \mathtt{hop\_counter} + 1 + \mathtt{hopcount}_d, \mathtt{timeout})\rangle,$
$\mathtt{out}\langle c_{\mathtt{Request\_Zone},r}, (\mathtt{rreq}, s, \mathit{Bid}, d, \mathtt{Request\_Zone}, \mathtt{seq\#}_s, (\mathtt{hop\_counter}) + 1)\rangle)).Q\_LAR1$

- $\mathtt{find\_path}(n)$ = $\mathtt{true}$ if there exists a valid path for n in the route table of the node executing the process;
- $\mathtt{newBid}$: generates a new unique *Bid* identifier for a packet;
- $\mathtt{lastBid}$: returns the latest generated *Bid* identifier;
- $\mathtt{control}(\mathit{Bid})$ = $\mathtt{true}$ if the request associated with *Bid* has been already received by the node executing the process.

Each node maintains a *routing table* containing information about the paths to the other nodes in the network. Each entry has the following form:

$$(d, \mathtt{seq\#}_d, \mathtt{next\_hop}_d, \mathtt{hopcount}_d, \mathtt{loc}_d, \mathtt{v}_d, \mathtt{timeout}),$$

where d is the destination name, $\mathtt{seq\#}_d$ is the sequence number of the route to d, $\mathtt{next\_hop}_d$ is the name of the next node to reach d, $\mathtt{hopcount}_d$ is the number of hops to reach d, $\mathtt{loc}_d$ is the last location known for d, $\mathtt{v}_d$ is the average speed of d and $\mathtt{timeout}$ is the timeout associated with the entry.

Each node is also associated with a *request table* containing the list of all the requests already processed by the node; this is needed to prevent loops during the route request forwarding. For brevity, we model a network in which all the nodes use a common transmission radius r.

Let us now consider $N = (\nu c)(n[P]_l \mid \prod_{i\in I} n_i[Q\_SIMPLE]_{l_i})$ where a node n broadcasts a route request using the simple flooding algorithm to find a path to m in the network $\prod_{i\in I} n_i$, and $M = (\nu c)(n[P]_l \mid \prod_{i\in I} n_i[Q\_LAR1]_{l_i})$ which is the same network but with nodes in I using the LAR protocol (Scheme 1) instead of the simple flooding algorithm.

The process executed by node n simply broadcasts a RREQ packet for node m and waits for a RREP packet until a timeout expires. The timeout is modelled using the operator $\oplus$ that behaves as the non-deterministic choice and can be implemented in our calculus by means of the parallel composition is the standard way. In case of timeout, a new RREQ is sent.

$$P = \mathtt{out}\langle c_{\emptyset,r}, (\mathtt{rreq}, n, \mathtt{newBid}, m, \mathtt{Request\_Zone}, \mathtt{seq\#}_n, 0)\rangle.P'$$
$$P' = P \oplus \mathtt{in}(c, x_1, x_2, x_3, x_4, x_5, x_6, x_7).[x_1 = \mathtt{rrep}][x_2 = n][x_3 = \mathtt{lastBid}] \times [x_4 = m][\mathtt{geq}(\mathtt{hop\_count}_m, x_7)]\mathtt{out}\langle \mathtt{ok}_{\mathtt{gps},r}, \mathtt{route\_found}\rangle, P'$$

where $m = n_i$ for some $i \in I$, and $x_7 = \mathtt{hop\_count}$ in the RREP packet received. Basically, once a route is found, n broadcasts on channel $\mathtt{ok}$ a packet that signals this event. Therefore, we consider that the two networks are probabilistic equivalent with respect to their ability to find a route to m if we observe this transmission with the same probability. Notice that, the output on channel c will not be observed by any location because we want to allow the route discovery packets used in the two networks to be arbitrary different.

Hereafter, we use $X \in \{\mathit{SIMPLE}, \mathit{LAR1}\}$ to denote the simple flooding or LAR Scheme 1. The *RREQ_SIMPLE* and the *RREQ_LAR*1 subprocess are defined as shown by Table 8.

In order to compare the behaviour of the protocols, we focus our attention on the following restricted set $\mathcal{F} \subseteq \mathit{Sched}$ of admissible schedulers:

1. the timeout for a RREQ identified by *Bid* occurs when in the networks there are no packets related to *Bid*;
2. nodes' movements are allowed at least every time a timeout occurs;
3. begin broadcasting actions (Beg-Bcast) have priority over end broadcasting actions (End-Bcast).

Condition 1 on $\mathcal{F}$ is a requirement inherited by the protocol design; the timeout is usually set by knowing the physical dimension of the network. Roughly speaking, we aim at preventing that in the analysis we consider unrealistic schedulers that always choose the timeout option too quickly and hence a route to the destination is never found and those schedulers that wait for an answer indefinitely long. Condition 2 is needed because we do not want to consider those schedulers that never allow for node movements. Finally, Condition 3 gives us the worst case scenario about the interference, i.e., whenever an interference could occur it is measured.

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

Proposition 6.1. *Let M, N and $\mathcal{F}$ as above. A sufficient condition for $M \approx_p^{\mathcal{F}} N$ is that the Markov chains $\mathbf{J}^{n_i}$ associated with the mobile nodes n_i ($i \in I$) are ergodic.*

For brevity we omit the formal proof. This relies on the fact that the probability of finding a route is always 1 both for the LAR and the flooding protocol. Indeed, node m keeps sending RREQ until it gets an answer thanks to the timeout mechanism that is eventually chosen by the hypothesis on $\mathcal{F}$. A route is surely eventually found thanks to the second assumption on the schedulers in $\mathcal{F}$ and the condition on the ergodicity of the chains modelling the nodes' movements (there is at least a node spatial configuration reachable with non zero probability in which the route from m to n is found without interference).

The comparison between LAR and flooding protocols in terms of interference must be carried out using PEBUM given the physical properties of the networks. Indeed, the interference levels can depend on several factors such as the node density and the good estimation of the *Request Zone* and the *Expected Zone* in the LAR.

7. Conclusion

One of the most critical challenges in managing mobile ad-hoc networks is to find a good trade off between network connectivity, power saving and interference reduction. We have proposed an effective framework for analyzing protocol connectivity and measuring the level of interference and, based on that for developing novel interference-aware communication strategies. Though other models exist in the literature, ours appears to be the most comprehensive and effective for the behavioural analysis and a quantitative assessment of interference for wireless networks in the presence of node mobility.

Plans for the future, include work on developing a model checker for our calculus based on PRISM [29,30] to perform automated, quantitative verification and analysis of wireless networks for a range of performance metrics. PRISM appears an excellent tool for the purpose, as it allows one to model process algebra operators, it supports models where non-deterministic and probabilistic aspects coexist, and provides support for the specification of a wide range of properties and rewards.

Acknowledgement

Work partially supported by the Italian MIUR - PRIN Project *CINA: Compositionality, Interaction, Negotiation and Autonomicity*. The work of Sardaouna Hamadou has been partially supported by the project ANR-09-BLAN-0169-01 PANDA: PArallel aNd Distributed Analysis, and the Inria large scale initiative CAPPRIS (Collaborative Action for the Protection of Privacy Rights in the Information Society).

Appendix

This supplement contains the proofs of some of the main results presented in the paper.

Proof of Lemma 3.1. By induction on the transition rules of Table 5.

Case 1: $M \xrightarrow{c?@l} [\![M']\!]_\Delta$

(Beg-Rcv) Let $M \xrightarrow{c?@l} [\![M']\!]_\Delta$ inferred by rule (Beg-Rcv), then there exist n, P, l, r such that $M \equiv n[P]_l$, $M' \equiv n[P']_l$ and $P = \mathtt{in}(c, \tilde{x}).Q$ and $P' = c(\tilde{x}).Q$. If we consider the empty network M_1 and the empty sequence $\tilde{d}$ the lemma is proved.

(Par) Let $M \xrightarrow{c?@l} [\![M']\!]_\Delta$ inferred by rule (Par), where $M \equiv M_1 \mid N$, $M' \equiv M_1' \mid N$ and $M_1 \xrightarrow{c?@l} [\![M_1']\!]_\Delta$. By induction hypothesis we have $M_1 \equiv (\nu\tilde{d})(n[\mathtt{in}(c, \tilde{x}).P]_l|M_2)$ and $M_1' \equiv (\nu\tilde{d})(n[c(\tilde{x}).P]_l|M_2)$, for some $n, P, \tilde{x}$, (possibly empty) $\tilde{d}$ such that $c \notin \tilde{d}$, and (possibly empty) M_2. By applying rule (Struct Cxt Par), (Struct Par Assoc), (Struct Res Par) and (Struct Trans) we obtain

$$M \equiv M_1 \mid N \equiv (\nu\tilde{d})(n[\mathtt{in}(c, \tilde{x}).P]_l|(M_2 \mid N))$$

and

$$M' \equiv M_1' \mid N \equiv (\nu\tilde{d})(n[\mathtt{in}(c, \tilde{x}).P]_l|(M_2 \mid N)),$$

as required.

(Res) Let $M \xrightarrow{c?@l} [\![M']\!]_\Delta$ inferred by rule (Res), where $M \equiv (\nu d)M_1$, $M' \equiv (\nu d)M_1'$ and $M_1 \xrightarrow{c?@l} [\![M_1']\!]_\Delta$ and $c \neq d$. By induction hypothesis we have $M_1 \equiv (\nu\tilde{d}')(n[\mathtt{in}(c, \tilde{x}).P]_l|M_2)$ and $M_1' \equiv (\nu\tilde{d}')(n[c(\tilde{x}).P]_l|M_2)$ for some $n, P, \tilde{x}$, (possibly empty) $\tilde{d}'$ such that $c \notin \tilde{d}'$, and (possibly empty) M_2. If we consider $\tilde{d}'' = \tilde{d}' \cup d$, since $c \notin \tilde{d}''$, we get:

$$M \equiv (\nu\tilde{d}'')(n[\mathtt{in}(c, \tilde{x}).P]_l|M_2) \quad \text{and} \quad M' \equiv (\nu\tilde{d}'')(n[\mathtt{in}(c, \tilde{x}).P]_l|M_2).$$

Case 2: $M \xrightarrow{c?\vartheta@l} [\![M']\!]_\Delta$. The proof of this case is analogous to the previous one.

Case 3: $M \xrightarrow{c_L![l,r]} [\![M']\!]_\Delta$

(Beg-Snd) Let $M \xrightarrow{c_L![l,r]} [\![M']\!]_\Delta$ inferred by rule (Beg-Snd). Then there exist $\tilde{v}$ and P such that: $M \equiv n[\mathtt{out}\langle c_{L,r}, \tilde{v}\rangle.P]_l$. Since $\mathtt{out}\langle c_{L,r}, \tilde{v}\rangle.P \xrightarrow{\bar{c}_{L,r}} \bar{c}_{L,r}\langle\tilde{v}\rangle.P$, if we suppose $\tilde{d}$, J, K and M_1 empty, lemma is proved because $M \equiv (\nu\tilde{d})(n[\mathtt{out}\langle c_{L,r}, \tilde{v}\rangle.P]_l | \prod_{j\in J} n_j[\mathtt{in}(c, \tilde{x_j})P_j]_{l_j} | \prod_{k\in K} n_k[c(\tilde{x_k})P_k]_{l_k} | M_1)$ and $M' \equiv (\nu\tilde{d})(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l | \prod_{j\in J} n_j[c(\tilde{x_j})P_j]_{l_j} | \prod_{k\in K} n_k[P_k\{\bot/\tilde{x_k}\}]_{l_k} \mid M_1)$.

(Beg-Bcast) Let $M \xrightarrow{c_L![l,r]} [\![M']\!]_\Delta$ because $M \equiv M_1 \mid N$, $M' \equiv M_1' \mid N'$, $M_1 \xrightarrow{c_L![l,r]} [\![M_1']\!]_\Delta$ and $N \xrightarrow{c?@l'} [\![N']\!]_\Delta$, with $d(l, l') \leq r$. By induction hypothesis: $M_1 \equiv (\nu\tilde{d_1})(n[\mathtt{out}\langle c_{L,r}, \tilde{v}\rangle.P]_l | \prod_{j\in J} n_j[\mathtt{in}(c, \tilde{x_j})P_j]_{l_j} | \prod_{k\in K} n_k[c(\tilde{x_k})P_k]_{l_k} | M_2)$ and $M_1' \equiv (\nu\tilde{d_1})(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l | \prod_{j\in J} n_j[c(\tilde{x_j})P_j]_{l_j} | \prod_{k\in K} n_k[P_k\{\bot/\tilde{x_k}\}]_{l_k} \mid M_2)$, for some n, P, $\tilde{v}$, l, some (possible empty) sequence $\tilde{d}_1$ such that $c \notin \tilde{d_1}$, some (possibly empty) sets J and K and some (possibly empty) network M_2, and by induction hypothesis we get:

$$N \equiv (\nu\tilde{d_2})(m[\mathtt{in}(c, \tilde{x}).Q]_{l'} | N_1) \quad \text{and} \quad N' \equiv (\nu\tilde{d_2})(m[c(\tilde{x}).Q]_{l'} | N_1),$$

for some m, Q, $\tilde{x}$, some (possible empty) sequence $\tilde{d_2}$ such that $c \notin \tilde{d_2}$ and (possibly empty) network N_1. By applying rules (Struct Cxt Par), (Struct Par Assoc), (Struct Res Par) and (Struct Trans), if we consider $\tilde{d} = \tilde{d_1} \cup \tilde{d_2}$, we get:

$$M \equiv (\nu\tilde{d})\left(n[\mathtt{out}\langle c_{L,r}, \tilde{v}\rangle.P]_l | m[\mathtt{in}(c, \tilde{x}).Q]_{l'} | \prod_{j\in J} n_j[\mathtt{in}(c, \tilde{x_j})P_j]_{l_j} | \prod_{k\in K} n_k[c(\tilde{x_k})P_k]_{l_k} \mid (M_2 \mid N_1)\right)$$

and

$$M' \equiv (\nu\tilde{d})\left(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l | m[c(\tilde{x}).Q]_{l'} \mid \prod_{j\in J} n_j[c(\tilde{x_j})P_j]_{l_j} | \prod_{k\in K} n_k[P_k\{\bot/\tilde{x_k}\}]_{l_k} \mid (M_2 \mid N_1)\right).$$

(Coll) Let $M \xrightarrow{c_L![l,r]} [\![M']\!]_\Delta$ because $M \equiv M_1 \mid N$, $M' \equiv M_1' \mid N$, $M_1 \xrightarrow{c_L![l,r]} [\![M_1']\!]_\Delta$ and $N \xrightarrow{c?\bot@l'} [\![N']\!]_\Delta$, with $d(l, l') \leq r$. By induction hypothesis: $M_1 \equiv (\nu\tilde{d_1})(n[\mathtt{out}\langle c_{L,r}, \tilde{v}\rangle.P]_l | \prod_{j\in J} n_j[\mathtt{in}(c, \tilde{x_j})P_j]_{l_j} | \prod_{k\in K} n_k[c(\tilde{x_k})P_k]_{l_k} | M_2)$ and $M_1' \equiv (\nu\tilde{d_1})(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l | \prod_{j\in J} n_j[c(\tilde{x_j})P_j]_{l_j} | \prod_{k\in K} n_k[P_k\{\bot/\tilde{x_k}\}]_{l_k} \mid M_2)$, for some n, P, some (possibly empty) sequence $\tilde{d_1}$ such that $c \notin \tilde{d_1}$, some (possibly empty) sets J and K and some (possibly empty) network M_2, and by induction hypothesis we get: $N \equiv (\nu\tilde{d_2})(m[c(\tilde{x}).Q]_{l'} | N_1)$ and $N' \equiv (\nu\tilde{d_2})(m[Q\{\bot/\tilde{x}\}]_{l'} | N_1)$, for some m, Q, $\tilde{x}$, some (possibly empty) sequence $\tilde{d_2}$ such that $c \notin \tilde{d_2}$ and (possibly empty) network N_1. By applying rules (Struct Cxt Par), (Struct Par Assoc), (Struct Res Par) and (Struct Trans), if we consider $\tilde{d} = \tilde{d_1} \cup \tilde{d_2}$ we get:

$$M \equiv (\nu\tilde{d})\left(n[\mathtt{out}\langle c_{L,r}, \tilde{v}\rangle.P]_l | \prod_{j\in J} n_j[\mathtt{in}(c, \tilde{x_j})P_j]_{l_j} | \prod_{k\in K} n_k[c(\tilde{x_k})P_k]_{l_k} \mid m[c(\tilde{x}).Q]_{l'} | (M_2 \mid N_1)\right)$$

and

$$M' \equiv (\nu\tilde{d})\left(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l | \prod_{j\in J} n_j[c(\tilde{x_j})P_j]_{l_j} | \prod_{k\in K} n_k[P_k\{\bot/\tilde{x_k}\}]_{l_k} m[Q\{\bot/\tilde{x}\}]_{l'} \mid (M_2 \mid N_1)\right).$$

The proof of the other cases is analogous to the first part of the lemma.

Case 4: $M \xrightarrow{c_L!\tilde{v}[l,r]} [\![M']\!]_\Delta$.

(End-Snd) Let $M \xrightarrow{c_L!\tilde{v}[l,r]} [\![M']\!]_\Delta$ inferred by rule (End-Snd), then there exists P such that $M \equiv n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l$. Since $\bar{c}_{L,r}\langle\tilde{v}\rangle.P \xrightarrow{\bar{c}_{L,r}\tilde{v}} P$, if we suppose J, $\tilde{d}$ and M_1 empty, lemma is proved because

$$M \equiv (\nu\tilde{d})\left(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l | \prod_{j\in J} n_j[c(\tilde{x_j})P_j]_{l_j} \mid M_1\right)$$

and

$$M' \equiv (\nu\tilde{d})\left(n[P]_l | \prod_{j\in J} n_j[P_j\{\tilde{v}/\tilde{x_j}\}]_{l_j} \mid M_1\right).$$

(End-Bcast) Let $M \xrightarrow{c_L!\tilde{v}[l,r]} [\![M']\!]_\Delta$ because $M \equiv M_1 \mid N$, $M' \equiv M_1' \mid N$, $M_1 \xrightarrow{c_L!\tilde{v}[l,r]} [\![M_1']\!]_\Delta$ and $N \xrightarrow{c?\tilde{v}@l'} [\![N']\!]_\Delta$, with $d(l, l') \leq r$. By induction hypothesis:

$$M_1 \equiv (\nu\tilde{d_1})\left(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l | \prod_{j\in J} n_j[c(\tilde{x_j})P_j]_{l_j} \mid M_2\right)$$

and

$$M_1' \equiv (\nu \tilde{d_1})\left(n[P]_l | \prod_{j\in J} n_j[P_j\{\tilde{v}/\tilde{x_j}\}]_{l_j} \mid M_2\right),$$

for some n, P, some (possibly empty) sequence $\tilde{d_1}$ such that $c \notin \tilde{d_1}$, some (possibly empty) set J and some (possibly empty) network M_2, and by induction hypothesis we get:

$$N \equiv (\nu \tilde{d_2})(m[c(\tilde{x}).Q]_{l'} | N_1) \quad \text{and} \quad N' \equiv (\nu \tilde{d_2})(m[Q\{\tilde{v}/\tilde{x}\}]_{l'} | N_1),$$

for some m, Q, $\tilde{x}$, some (possibly empty) sequence $\tilde{d_2}$ such that $c \notin \tilde{d_2}$ and (possibly empty) network N_1. By applying rules (Struct Cxt Par), (Struct Par Assoc), (Struct Res Par) and (Struct Trans), if we consider $\tilde{d} = \tilde{d_1} \cup \tilde{d_2}$ we get:

$$M \equiv (\nu \tilde{d})\left(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l | m[c(\tilde{x}).Q]_{l'} \mid \prod_{j\in J} n_j[c(\tilde{x_j})P_j]_{l_j} \mid (M_1 \mid N_1)\right)$$

and

$$M' \equiv (\nu \tilde{d})\left(n[P]_l | m[Q\{\tilde{v}/\tilde{x}\}]_{l'} \mid \prod_{j\in J} n_j[P_j\{\tilde{v}/\tilde{x}\}]_{l_j} \mid (M_1 \mid N_1)\right).$$

The proof of the other cases is analogous to the first part of the lemma. □

Proof of Theorem 3.3. 1. By induction on the reduction $M \to [\![M']\!]_\theta$. Suppose that $M \to [\![M']\!]_\theta$ is due to the application of the rule (R-Move). We deduce $M \equiv M' \equiv n[P]_l$ and $\theta = \mu_l^n$, for some name n, some location l and some process P. We simply apply (Move) to obtain:

$$\frac{\texttt{Active}(P) = \texttt{false}}{n[P]_l \xrightarrow{\tau} [\![n[P]_l]\!]_{\mu_l^n}}.$$

Suppose that $M \to [\![M']\!]_\theta$ is due to the application of the rule (R-Par). If we consider $M \equiv M_1 \mid M_2$ and $M' \equiv M_1' \mid M_2$ we have:

$$\frac{M_1 \to [\![M_1']\!]_\theta}{M_1 \mid M_2 \to [\![M_1' \mid M_2]\!]_\theta}.$$

By induction hypothesis $M \xrightarrow{\tau} [\![M']\!]_\theta$, then by rule (Par) we get:

$$\frac{M_1 \xrightarrow{\tau} [\![M_1']\!]_\theta}{M_1 \mid M_2 \xrightarrow{\tau} [\![M_1' \mid M_2]\!]_\theta}.$$

Suppose that $M \to [\![M']\!]_\theta$ is due to the application of the rule (R-Res). If we consider $M \equiv (\nu c)M_1$ and $M' \equiv (\nu c)M_1'$ we have

$$\frac{M_1 \to [\![M_1']\!]_\theta}{(\nu c)M_1 \to [\![(\nu c)M_1']\!]_\theta},$$

by induction hypothesis $M \xrightarrow{\tau} [\![M']\!]_\theta$, then by applying rule (Res), since $\texttt{Chan}(\tau) \neq c$ we get:

$$\frac{M_1 \xrightarrow{\tau} [\![M_1']\!]_\theta}{(\nu c)M_1 \xrightarrow{\tau} [\![(\nu c)M_1']\!]_\theta}.$$

Suppose that $M \to [\![M']\!]_\theta$ is due to the application of the rule (R-Bgn-Bcast). It means:

$$M \equiv (\nu \tilde{d})\left(n[\texttt{out}\langle c_{L,r}, \tilde{v}\rangle.P]_l \mid \prod_{i\in I} n[\bar{c}_{L_i,r_i}\langle\tilde{v}_i\rangle.P_i]_{l_i} \mid \prod_{j\in J} n[c(x_j).P_j]_{l_j} \mid \prod_{k\in K} n[\texttt{in}(c, x_k).P_k]_{l_k}\right)$$

and

$$M' \equiv (\nu \tilde{d})\left(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid \prod_{i\in I} n[\bar{c}_{L_i,r_i}\langle\tilde{v}_i\rangle.P_i]_{l_i} \mid \prod_{j\in J} n[P_j\{\perp/\tilde{x}_j\}]_{l_j} \mid \prod_{k\in K} n[c(x_k).P_k]_{l_k}\right),$$

for some n, some process P, some channel c, some set L of locations, some radius r, some tuple $\tilde{v}$ of messages, some tuple $\tilde{d}$ of channels, and some (possibly empty) sets I, J and K of networks. Then, by applying rule (Beg-Snd), (Beg-Rcv), and then $|K|$ times rule (Beg-Bcast), $|J|$ times rule (Coll-Bcast), and, finally rules (Res), (Lose1) and (Par), we obtain: $M \xrightarrow{\tau} \equiv [\![(\nu \tilde{d})(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid \prod_{i\in I} n[\bar{c}_{L_i,r_i}\langle\tilde{v}_i\rangle.P_i]_{l_i} \mid \prod_{j\in J} n[P_j\{\perp/\tilde{x}_j\}]_{l_j} \mid \prod_{k\in K} n[c(x_k).P_k]_{l_k})]\!]_\theta$ as required.

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

Suppose that $M \to [\![M']\!]_\theta$ is due to the application of the rule (R-End-Bcast). It means:

$$M \equiv (\nu \tilde{d})n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid \prod_{j\in J} n_j[c(\tilde{x}_j).P_j]_{l_j}$$

and

$$M' \equiv (\nu \tilde{d})n[P]_l \mid \prod_{j\in J} n_j[P_j\{\tilde{v}/\tilde{x}_j\}]_{l_j}$$

for some channel c, some tuple $\tilde{d}$ of channels such that $c \notin \tilde{d}$, some node n, some process P, some tuple $\tilde{v}$ of messages, some location l, some set L of locations, some radius r, some process P, and some (possibly empty set) J such that $d(l, l_i) \leq r\ \forall i \in J$.
Then, by applying rule (End-Snd), (End-Rcv), $|I|$ times rule (End-Bcast), $|\tilde{d}|$ times (Res) and finally rule (Lose2), we get:

$$(\nu \tilde{d})n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid \prod_{j\in J} n_j[c(\tilde{x}_j).P_j]_{l_j} \xrightarrow{\tau} \left[\!\!\left[(\nu \tilde{d})n[P]_l \mid \prod_{j\in J} n_j[P_j\{\tilde{v}/\tilde{x}_j\}]_{l_j} \right]\!\!\right]_\Delta$$

as required.
Finally let suppose that the reduction $M \to [\![M']\!]_\theta$ is due to an application of rule (R-Struct):

$$\frac{M \equiv N \quad N \to [\![N']\!]_\theta \quad N' \equiv M'}{M \to [\![M']\!]_\theta}.$$

By induction hypothesis there exists $N_1 \equiv N$ and $N_2 \equiv N'$ such that $N_1 \xrightarrow{\tau} [\![N_2]\!]_\theta$. Then, by applying the rule for structural congruence (Struct Trans) we get $M \equiv N \equiv N_1$ and $M' \equiv N' \equiv N_2$, as required.

2. The second point of the theorem follows from Lemma 3.1 and the definition of barb. If $M\downarrow_{c@K}$, by definition of barb there exists $\tilde{v}$, L, r, a (possibly empty) sequence $\tilde{d}$ such that $c \notin \tilde{d}$, a process P, a (possibly empty) network M_1 such that: $M \equiv (\nu \tilde{d})(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid M_1)$, with $K \subseteq \{k \in L \text{ s.t. } d(l,k) \leq r\}$ and $K \neq \emptyset$.
By applying the rules (End-Snd) and (Par) we obtain:

$$\frac{n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \xrightarrow{c_L!\tilde{v}[l,r]} [\![n[P]_l]\!]_\Delta}{n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid M_1 \xrightarrow{c_L!\tilde{v}[l,r]} [\![n[P]_l \mid M_1]\!]_\Delta};$$

then, since $K \subseteq R \cap L$ and $K \neq \emptyset$, we can apply rule (Obs):

$$n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid M_1 \xrightarrow{c!\tilde{v}@K\lhd R} [\![n[P]_l \mid M_1]\!]_\Delta,$$

where $R = \{l' \in \mathbf{Loc} : d(l,l') \leq r\}$, as required.
If $M \xrightarrow{c!\tilde{v}@K\lhd R} [\![M']\!]_\Delta$, because $M \xrightarrow{c_L!\tilde{v}[l,r]} [\![M']\!]_\Delta$, by applying Lemma 3.1 then there exist n, a (possibly empty) sequence $\tilde{d}$ such that $c \notin \tilde{d}$, P, a (possibly empty) network M_1 and a (possibly empty) set J, such that $\forall j \in J\ d(l, l_j) \leq r$ and:

$$M \equiv (\nu \tilde{d}) \left(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid \prod_{j\in J} n_j[c(\tilde{x}_j).P_j]_{l_j} \mid M_1 \right)$$

and

$$M \equiv (\nu \tilde{d}) \left(n[P]_l \mid \prod_{j\in J} n_j[P_j\{\tilde{v}/\tilde{x}_j\}]_{l_j} \mid M_1 \right).$$

By applying the definition of barb we conclude $M\downarrow_{c@K}$.

3. The third point of the theorem is proved by induction on the derivation $M \xrightarrow{\tau} [\![M']\!]_\theta$. Suppose that $M \xrightarrow{\tau} [\![M']\!]_\theta$ is due to an application of the rule (Move), that means:

$$\frac{\texttt{Active}(P) = \texttt{false}}{n[P]_l \xrightarrow{\tau} [\![n[P]_l]\!]_{\mu_l^n}},$$

hence, by applying (R-Move) we get:

$$\frac{\texttt{Active}(P) = \texttt{false}}{n[P]_l \to [\![n[P]_l]\!]_{\mu_l^n}}.$$

If $M \xrightarrow{\tau} [\![M']\!]_\theta$ is due to an application of (Lose1):

$$\frac{M \xrightarrow{c_L![l,r]} [\![M']\!]_\Delta}{M \xrightarrow{\tau} [\![M']\!]_\Delta},$$

then, by applying Lemma 3.1, there exists n, $\tilde{v}$, P $\tilde{d}$ such that $c \notin \tilde{d}$ and P, a (possibly empty) network M_1 and two (possibly empty) sets J and K such that $\forall i \in J \cup K\ d(l, l_i) \leq r$, such that:

$$M \equiv (\nu\tilde{d})\left(n[\mathtt{out}\langle c_{L,r}, \tilde{v}\rangle.P]_l | \prod_{j\in J} n_j[\mathtt{in}(c, \tilde{x}_j).P_j]_{l_j} \mid \prod_{k\in K} n_k[c(\tilde{x}_k).P_j]_{l_j} \mid M_1\right)$$

$$M' \equiv (\nu\tilde{d})\left(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l | \prod_{j\in J} n_j[c(\tilde{x}_j).P_j]_{l_j} | \prod_{k\in K} n_k[P_k\{\perp/\tilde{x}_k\}]_{l_k} | M_1\right).$$

Finally, by applying rule (R-Bgn-Bcast), (R-Res) and (R-Struct) we get $M \to [\![M']\!]_\theta$.

For the application of the rule (Lose2) the proof is analogous to the previous one.

Suppose that $M \xrightarrow{\tau} [\![M']\!]_\theta$ is due to the application of (Res), we have $M \equiv (\nu c)M_1$, $M' \equiv (\nu c)M_1'$ and

$$\frac{M_1 \xrightarrow{\tau} [\![M_1']\!]_\theta}{(\nu c)M_1 \xrightarrow{\tau} [\![(\nu c)M_1']\!]_\theta}.$$

By induction hypothesis $M_1 \to [\![M_1']\!]_\theta$, hence, by applying rule (R-Res) we get $(\nu c)M_1 \to [\![(\nu c)M_1']\!]_\theta$.

Finally, suppose that $M \xrightarrow{\tau} [\![M']\!]_\theta$ is due to the application of (Par), we have $M \equiv M_1 \mid N$, $M' \equiv M_1' \mid N$ and:

$$\frac{M_1 \xrightarrow{\tau} [\![M_1']\!]_\theta}{M_1|N \xrightarrow{\tau} [\![M_1'|N]\!]_\theta},$$

by induction hypothesis $M_1 \to [\![M_1']\!]_\theta$, hence, by applying rule (R-Par) we get $M_1|N \to [\![M_1'|N]\!]_\theta$.

4. The last point of the theorem follows from definition of barb and Lemma 3.1. Formally, since $M \xrightarrow{c!\tilde{v}@K\triangleleft R} [\![M']\!]_\Delta$ because $M \xrightarrow{c_L!\tilde{v}[l,r]} [\![M']\!]_\Delta$ for some location l, radius r and set L of intended recipients, by applying Lemma 3.1:

$$M \equiv (\nu\tilde{d})\left(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid \prod_{j\in J} n_j[c(\tilde{x}_j).P_j]_{l_j} \mid M_1\right)$$

and

$$M' \equiv (\nu\tilde{d})\left(n[P]_l \mid \prod_{j\in J} n_j[P_j\{\tilde{v}/\tilde{x}_j\}]_{l_j} \mid M_1\right)$$

for some n, P, for some (possibly empty) sequence $\tilde{d}$ such that $c \notin \tilde{d}$, some (possibly empty) set J, and some (possibly empty) network M_1. Then, by applying the rule (R-End-Bcast), (R-Par) and (R-Res) we get

$$(\nu\tilde{d})\left(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid \prod_{j\in J} n_j[c(\tilde{x}_j).P_j]_{l_j} \mid M_1\right) \to \left[\!\!\left[(\nu\tilde{d})\left(n[P]_l \mid \prod_{j\in J} n_j[P_j\{\tilde{v}/\tilde{x}_j\}]_{l_j} \mid M_1\right)\right]\!\!\right]_\Delta,$$

and, by applying (R-Struct), we obtain $M \to [\![M']\!]_\Delta$, as required. □

Proof of Theorem 3.10. In order to prove that bisimulation is a sound characterization of Probabilistic Barbed Congruence we have to prove that $\approx_p^{\hat{\mathcal{F}}}$ is:

1. reduction closed w.r.t. $\mathcal{F}$
2. probabilistic barb preserving w.r.t. $\mathcal{F}$
3. contextual.

Probabilistic Labelled Bisimulation is reduction closed.

We have to prove that if $M \approx_p^{\mathcal{F}} N$, then for all $F \in \mathcal{F}_{\mathcal{C}}^M$, there exists $F' \in \mathcal{F}_{\mathcal{C}}^N$ such that for all classes $\mathcal{C} \in \mathcal{N}/\mathcal{R}$, $Prob_M^F(\mathcal{C}) = Prob_N^{F'}(\mathcal{C})$.

By Theorem 3.3 there exists an admissible scheduler $\hat{F} \in LSched$ such that $Prob_M^F(\mathcal{C}) = Prob_M^{\hat{F}}(\Longrightarrow, \mathcal{C}')$, where $\mathcal{C}' = \mathcal{C} \cup \{M' : M' \equiv M'' \in \mathcal{C}\}$, but since $\forall M'$ such that $M' \equiv M'' \in \mathcal{C}$, by applying rule (R-Struct) it holds $M' \cong_p^{\mathcal{F}} M''$, we get $\{M' : M' \equiv M'' \in \mathcal{C}\} \subseteq \mathcal{C}$, that means $\mathcal{C}' = \mathcal{C}$. By Definition 3.5 we deduce $\hat{F} \in \hat{\mathcal{F}}_{\mathcal{C}}^M$, since for all the executions in $Exec_M^{\hat{F}}(\Longrightarrow, \mathcal{C})$, the correspondent reduction executions are allowed by F, which is an element of $\mathcal{F}_{\mathcal{C}}^M$. By Proposition 3.9 we have that $\exists \hat{F}' \in \hat{\mathcal{F}}_{\mathcal{C}}^N$ such that $Prob_M^{\hat{F}}(\Longrightarrow, \mathcal{C}) = Prob_N^{\hat{F}'}(\Longrightarrow, \mathcal{C})$. Finally, by Theorem 3.3, $\exists F' \in Sched$ such that $Prob_N^{\hat{F}'}(\Longrightarrow, \mathcal{C}) = Prob_N^{F'}(\mathcal{C})$. Finally, we can deduce $F' \in \mathcal{F}_{\mathcal{C}}^N$ by applying Definitions 3.5 and 2.7.

Probabilistic Labelled Bisimulation is Probabilistic barb preserving.

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

To prove that bisimulation is probabilistic barb preserving we have to show that, if $M \approx_p^{\mathcal{F}} N$, then, for each scheduler $F \in \mathcal{F}_{\mathcal{C}}^{M}$, for each channel c, and for each set K of locations such that $M \Downarrow_p^F c@K$, then $\exists F' \in \mathcal{F}_{\mathcal{C}}^{N}$ such that $N \Downarrow_p^{F'} c@K$.

Let $M \Downarrow_p^F c@K$ for some channel c, some set K of locations, and scheduler $F \in \mathcal{F}_{\mathcal{C}}^{M}$. It means that $Prob_M^F(H) = p$, where $M' \in H$ iff $M' \downarrow_{c@K}$. We can partition H in a set of equivalence classes for $\approx_p^{\mathcal{F}}$. Hence $\exists I$ such that $\forall i \in I\ \mathcal{C}_i \in \mathcal{N}/\approx_p^{\mathcal{F}}$, $\mathcal{C}_i \cap H \neq \emptyset$, and $H \subseteq \bigcup_{i\in I} \mathcal{C}_i$. We get:

$$Prob_M^F(H) = \sum\nolimits_{e\in Exec_M^F(H)} P_M^F(e) = \sum_{i\in I} Prob_M^F(\mathcal{C}_i) = p.$$

By Theorem 3.3, $\exists \hat{F} \in LSched$ such that $\forall i \in I$:

$$Prob_M^F(\mathcal{C}_i) = Prob_M^{\hat{F}}(\Longrightarrow, \mathcal{C}_i'),$$

where $\mathcal{C}_i' = \mathcal{C}_i \cup \{M' \mid \exists M'' \in \mathcal{C}_i \text{ and } M' \equiv M''\}$, but, since $\cong_p^{\mathcal{F}}$ is closed under structural congruence, $\forall M' \equiv M'' \in \mathcal{C}_i$, $M' \cong_p^{\mathcal{F}} M''$, hence $\{M' : M' \equiv M'' \in \mathcal{C}_i\} \subseteq \mathcal{C}_i$, that means $\mathcal{C}_i' = \mathcal{C}_i$. Now we have to prove that $\hat{F} \in \hat{\mathcal{F}}_{\mathcal{C}}^{M}$, but this follows straightforwardly by Definition 3.5. Hence:

$$Prob_M^F(\mathcal{C}_i) = Prob_M^{\hat{F}}(\Longrightarrow, \mathcal{C}_i) \forall i \in I \quad \text{and}$$
$$\sum_{i\in I} Prob_M^F(\mathcal{C}_i) = \sum_{i\in I} Prob_M^{\hat{F}}(\Longrightarrow, \mathcal{C}_i).$$

Since $M \approx_p^{\mathcal{F}} N$, $\exists \hat{F}' \in \hat{\mathcal{F}}_{\mathcal{C}}^{N}$ such that, $\forall i \in I$:

$$Prob_M^{\hat{F}}(\Longrightarrow, \mathcal{C}_i) = Prob_N^{\hat{F}'}(\Longrightarrow, \mathcal{C}_i).$$

Again by Theorem 3.3 $\exists F' \in LSched$ such that:

$$p = \sum_{i\in I} Prob_N^{\hat{F}'}(\Longrightarrow, \mathcal{C}_i) = \sum_{i\in I} Prob_N^{F'}(\mathcal{C}_i) = Prob_N^{F'}(H),$$

that means $N \Downarrow_p^{F'} c@K$.

By Definition 3.5 we finally deduce that $F' \in \mathcal{F}_{\mathcal{C}}^{N}$, as required.

Probabilistic Labelled Bisimulation is contextual

We start with the parallel composition. Let $\mathcal{R}$ be the following relation:

$$\mathcal{R} = \{(M \mid O, N \mid O) : M, N, M \mid O, N \mid O \text{ are well-formed, and, } M \approx_p^{\mathcal{F}} N\}.$$

We will prove that it is a probabilistic labelled bisimulation w.r.t. $\mathcal{F}$. For this purpose, we need to prove that, $\forall F \in \hat{\mathcal{F}}_{\mathcal{C}}^{M|O}$ $\exists F' \in \hat{\mathcal{F}}_{\mathcal{C}}^{N|O}$ such that, $\forall \mathcal{C} \in \mathcal{N}/\mathcal{R}$, $\forall \alpha$:

1. $\alpha = \tau$ then $Prob_{M|O}^F(\xrightarrow{\tau}, \mathcal{C}) = Prob_{N|O}^{F'}(\Longrightarrow, \mathcal{C})$.

   If $P, Q \in \mathcal{C}$, then, by definition of $\mathcal{R}$, $P \equiv \bar{P} \mid \bar{O}$, $Q \equiv \bar{Q} \mid \bar{O}$ and $\bar{P} \approx_p^{\mathcal{F}} \bar{Q}$. But then there exists $\mathcal{D} \in \mathcal{N}/\approx_p^{\mathcal{F}}$ such that $\mathcal{D} = \{\bar{P} : \bar{P} \mid \bar{O} \in \mathcal{C}\}$. Now we have three cases to consider:

   (i) if $M \mid O \xrightarrow{\tau} [\![M \mid O']\!]_\theta$ the proof is simple, because we have, $\forall \bar{M}$ in the support of $[\![M \mid O']\!]_\theta$, such that $\bar{M} \in \mathcal{C}$, $\bar{M} \equiv M \mid O''$ and, since $M \approx_p^{\mathcal{F}} N$, $N \mid O'' \in \mathcal{C}$ too, by definition of $\mathcal{R}$. Hence (by applying rule (Par) to the action $O \xrightarrow{\tau} [\![O']\!]_\theta$), since $N \mid O$ is well-formed, $\exists F' \in LSched$ such that
   $$Prob_{M|O}^F(\xrightarrow{\tau}, \mathcal{C}) = Prob_{N|O}^{F'}(\Longrightarrow, \mathcal{C}).$$
   We have only to prove that $F' \in \hat{\mathcal{F}}_{\mathcal{C}}^{N|O}$, but the proof follows straightforwardly by the Definitions 2.7 and 3.5.

   (ii) If $M \mid O \xrightarrow{\tau} [\![M' \mid O]\!]_\theta$, since M is well-formed, by Definition 3.5 $\exists F_1 \in \hat{\mathcal{F}}_{\mathcal{C}}^{M}$ such that $Prob_{M|O}^F(\xrightarrow{\tau}, \mathcal{C}) = Prob_M^{F_1}(\xrightarrow{\tau}, \mathcal{D})$. But since $M \approx_p^{\mathcal{F}} N$, and N is well-formed, $\exists F_2 \in \hat{\mathcal{F}}_{\mathcal{C}}^{N}$ such that $Prob_M^{F_1}(\xrightarrow{\tau}, \mathcal{D}) = Prob_N^{F_2}(\Longrightarrow, \mathcal{D})$. Again, since the network $N \mid O$ is well-formed, $\exists F' \in LSched$ such that, by applying rule (Par) to the executions in $Exec_N^{F_2}(\Longrightarrow, \mathcal{D})$, we get
   $$Prob_N^{F_2}(\Longrightarrow, \mathcal{D}) = Prob_{N|O}^{F'}(\Longrightarrow, \mathcal{C}).$$
   Since by Definition 3.5 each execution in the set $Exec_N^{F_2}(\Longrightarrow, \mathcal{D})$ has a correspondent reduction execution allowed by $\mathcal{F}_{\mathcal{C}}^{N|O}$, and by Definition 2.7 we know that the same executions can be performed by N when interacting with any context, we can finally deduce, by applying again Definition 3.5, that $F' \in \hat{\mathcal{F}}_{\mathcal{C}}^{N|O}$, as required.

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

(iii) If $M \mid O \xrightarrow{\tau} M' \mid O'$ due to a synchronization between M and O, then there are two cases to consider.

If $M \xrightarrow{c!\tilde{v}[l,r]} [\![M']\!]_{\Delta}$ and $O \xrightarrow{c?\tilde{v}@k} [\![O']\!]_{\Delta}$, for some message $\tilde{v}$, channel c, locations l, k and radius r, such that $d(l,k) \leq r$, we can apply rule (Obs) obtaining $M \xrightarrow{c!\tilde{v}@K \triangleleft R} M'$ for some $K \subseteq L$ and for some R, with $k \in R$. Therefore, $\exists F_1 \in LSched$ such that:

$$Prob^{F}_{M|O}\left(\xrightarrow{\tau}, \mathcal{C}\right) = Prob^{F_1}_{M}\left(\xrightarrow{c!\tilde{v}@K \triangleleft R}, \mathcal{D}\right).$$

By Definition 3.5 we deduce $F_1 \in \hat{\mathcal{F}}^{M}_{\mathcal{C}}$ and, since $N \approx^{\mathcal{F}}_{p} M$, $\exists F_2 \in \hat{\mathcal{F}}^{N}_{\mathcal{C}}$ such that

$$Prob^{F_1}_{M}\left(\xrightarrow{c!\tilde{v}@K \triangleleft R}, \mathcal{D}\right) = Prob^{F_2}_{N}\left(\stackrel{c!\tilde{v}@K \triangleleft R}{\Longrightarrow}, \mathcal{D}\right),$$

where each execution e in $Exec^{F_2}_{N}(\stackrel{c!\tilde{v}@K \triangleleft R}{\Longrightarrow}, \mathcal{D})$ is of the form

$$e = N \xrightarrow{\tau}_{\theta_1} N_1 \rightarrow \cdots N_{i-1} \xrightarrow{c!\tilde{v}@K \triangleleft R}_{\Delta} N_i \rightarrow \cdots N',$$

and, by applying rule (Obs) backwardly, $N_{i-1} \xrightarrow{c!\tilde{v}[l',r']}_{\Delta} N_i$ for some l' and r' such that $d(l',k) \leq r'$. We can apply rule (Bcast) obtaining $N_{i-1} \mid O \xrightarrow{c!\tilde{v}[l',r']}_{\Delta} N_i \mid O'$ without changing probability. Finally if we take $F' \in LSched$ which applies rule (Lose2) to the output action, we obtain the required result:

$$Prob^{F_2}_{N}\left(\stackrel{c!\tilde{v}@K \triangleleft R}{\Longrightarrow}, \mathcal{D}\right) = Prob^{F'}_{N|O}(\Longrightarrow, \mathcal{C}).$$

We have finally to prove that $F' \in \hat{\mathcal{F}}^{N|O}_{\mathcal{C}}$. We start by the consideration that, by Theorem 3.3, for any execution of the form $\stackrel{\alpha}{\Longrightarrow}$ in $\hat{\mathcal{F}}^{N}_{\mathcal{C}}$, where α is a silent or an output action there exists a correspondent reduction in $\mathcal{F}^{N|O}_{\mathcal{C}}$. Since by Definition 2.7, for any context, there exists a scheduler in $\mathcal{F}^{N|O}_{\mathcal{C}}$ mimicking the behaviour exhibited by N when interacting with the given context, we can affirm that $\exists \bar{F} \in \mathcal{F}^{N|O}_{\mathcal{C}}$ such that $Exec^{\bar{F}}_{N|O}$ contains all the reductions corresponding to the executions of $Exec^{F'}_{N|O}$. Hence, by Definition 3.5, $F' \in \hat{\mathcal{F}}^{N|O}_{\mathcal{C}}$, as required.

If $M \xrightarrow{c?\tilde{v}@k} [\![M']\!]_{\Delta}$ and $O \xrightarrow{c_L!\tilde{v}[l,r]} [\![O']\!]_{\Delta}$, for some message $\tilde{v}$, some set L of locations, some channel c, some locations l, k and radius r, such that $d(l,k) \leq r$, then $\exists F_1 \in \hat{\mathcal{F}}^{M}_{\mathcal{C}}$ such that:

$$Prob^{F}_{M|O}(\xrightarrow{\tau}, \mathcal{C}) = Prob^{F_1}_{M}(\xrightarrow{c?\tilde{v}@k}, \mathcal{D}).$$

Since $N \approx^{\mathcal{F}}_{p} M$, $\exists F_2 \in \hat{\mathcal{F}}^{N}_{\mathcal{C}}$ such that:

$$Prob^{F_1}_{M}(\xrightarrow{c?\tilde{v}@k}, \mathcal{D}) = Prob^{F_2}_{N}(\stackrel{c?\tilde{v}@k}{\Longrightarrow}, \mathcal{D}) \quad \text{or}$$

$$Prob^{F_1}_{M}(\xrightarrow{c?\tilde{v}@k}, \mathcal{D}) = Prob^{F_2}_{N}(\Longrightarrow, \mathcal{D}).$$

In the first case, since by hypothesis $k \in R$ and $N \mid O$ is well-formed, also N is able to synchronize with O. Hence $\exists F' \in LSched$ such that for all

$$e = N \xrightarrow{\tau}_{\theta_1} N_1 \rightarrow \cdots N_{i-1} \xrightarrow{c?\tilde{v}@k} N_i \rightarrow \cdots N' \in Exec^{F_2}_{N}(\stackrel{c?\tilde{v}@k}{\Longrightarrow}, \mathcal{D})$$

there exists a matching execution such that, by applying rule (Bcast) $N_{i-1} \mid O \xrightarrow{c!\tilde{v}[l,r]} N_i \mid O$, and by applying rule (Lose2), we get:

$$e' = N \mid O \xrightarrow{\tau}_{\theta_1} N_1 \mid O \rightarrow \cdots N_{i-1} \mid O \xrightarrow{\tau} N_i \mid O' \rightarrow \cdots N' \mid O'$$

in $Exec^{F'}_{N|O}(\Longrightarrow, \mathcal{C})$. Hence,

$$Prob^{F_2}_{N}\left(\stackrel{c?\tilde{v}@k}{\Longrightarrow}, \mathcal{D}\right) = Prob^{F'}_{N|O}(\Longrightarrow, \mathcal{C}).$$

In order to prove $F' \in \hat{\mathcal{F}}^{N|O}_{\mathcal{C}}$, we start by the consideration that, since $O \xrightarrow{c_L!\tilde{v}[l,r]} [\![O']\!]_{\Delta}$, by Definition 2.7, for any context, there exists a scheduler in $\mathcal{F}^{N|O}_{\mathcal{C}}$ mimicking the behaviour of O in its interaction with the given context. Then we can affirm that $\exists \bar{F} \in \mathcal{F}^{N|O}_{\mathcal{C}}$ such that $Exec^{\bar{F}}_{N|O}$ contains all the reductions corresponding to the executions of $Exec^{F'}_{N|O}$. Hence, by Definition 3.5, $F' \in \hat{\mathcal{F}}^{N|O}_{\mathcal{C}}$, as required.

If N is not able to receive the message the proof is analogous: it is sufficient to apply the rule (Par) to $O \xrightarrow{c!\tilde{v}@K \triangleleft R} [\![O']\!]_{\Delta}$, obtaining:

$$Prob^{F_2}_{N}(\Longrightarrow, \mathcal{D}) = Prob^{F'}_{N|O}(\Longrightarrow, \mathcal{C}).$$

2. $\alpha = c!\tilde{v}@K \triangleleft R$

The proof is analogous to the point (iii) of the previous item.

3. $\alpha = c?@k$ then $Prob^{F}_{M|O}(\xrightarrow{\alpha}, \mathcal{C}) = Prob^{F'}_{N|O}(\stackrel{\alpha}{\Longrightarrow}, \mathcal{C})$ or $Prob^{F}_{M|O}(\xrightarrow{\alpha}, \mathcal{C}) = Prob^{F'}_{N|O}(\Longrightarrow, \mathcal{C})$.

If $P, Q \in \mathcal{C}$, then $P \equiv \bar{M} \mid \bar{O}$, $Q \equiv \bar{N} \mid \bar{O}$ and $\bar{M} \approx^{\mathcal{F}}_{p} \bar{N}$. But then $\exists \mathcal{D} \in \mathcal{N}/\approx^{\mathcal{F}}_{p}$ such that $\mathcal{D} = \{\bar{M} : \bar{M} \mid \bar{O} \in \mathcal{C}\}$. Now we have two cases to consider:

(i) The transition is due to an action performed by O, hence $O \xrightarrow{\alpha}_{\Delta} O'$ and $M \mid O' \in \mathcal{C}$. But since $M \approx^{\mathcal{F}}_{p} N$, $N \mid O' \in \mathcal{C}$ too, $\exists F' \in LSched$ such that by applying parallel composition to the input of O, we obtain the desired result:

$$Prob^{F}_{M|O}(\xrightarrow{\alpha}, \mathcal{C}) = Prob^{F'}_{N|O}(\stackrel{\alpha}{\Longrightarrow}, \mathcal{C}).$$

Finally, by Definition 3.5 we deduce $F' \in \hat{\mathcal{F}}_{\mathcal{C}}^{N|O}$, as required.

(ii) The transition is due to an action performed by M, in this case, by Definition 3.5 $\exists F_1 \in \hat{\mathcal{F}}_{\mathcal{C}}^{M}$ such that:

$$Prob_{M|O}^{F}(\xrightarrow{\alpha}, \mathcal{C}) = Prob_{M}^{F_1}(\xrightarrow{\alpha}, \mathcal{D}).$$

Since $M \approx_p^{\mathcal{F}} N$ $\exists F_2 \in \hat{\mathcal{F}}_{\mathcal{C}}^{N}$ such that:

$$Prob_{M}^{F_1}(\xrightarrow{\alpha}, \mathcal{D}) = Prob_{N}^{F_2}(\overset{\alpha}{\Longrightarrow}, \mathcal{D}), \quad \text{or}$$
$$Prob_{M}^{F_1}(\xrightarrow{\alpha}, \mathcal{D}) = Prob_{N}^{F_2}(\Longrightarrow, \mathcal{D}).$$

In both cases, since $N \mid O$ is well-formed, $\exists F' \in LSched$ such that by applying parallel composition, we have:

$$Prob_{N}^{F_2}(\overset{\alpha}{\Longrightarrow}, \mathcal{D}) = Prob_{N|O}^{F'}(\overset{\alpha}{\Longrightarrow}, \mathcal{C}), \quad \text{or}$$
$$Prob_{N}^{F_2}(\Longrightarrow, \mathcal{D}) = Prob_{N|O}^{F'}(\Longrightarrow, \mathcal{C}).$$

In order to prove that $F' \in \hat{\mathcal{F}}_{\mathcal{C}}^{N|O}$, we start by the consideration that, by Definition 3.5 there exists at least a context $C[\cdot]$ and $\exists \bar{F} \in \mathcal{F}_{\mathcal{C}}^{C[N]}$ such that $C[N] \to C'[N']$, and, by the reduction rules we get:

$$C[\cdot] \equiv (\nu \tilde{d}) m[\mathtt{out}\langle c_{L,r}, \tilde{v}\rangle.P]_l \mid M_1$$

for some $\tilde{d}$ such that $c \notin \tilde{d}$, some m, some set L of locations, some process P, some (possibly empty) network M_1, some location l and some radius r such that $d(l, k) \leq r$. Then, by Definition 2.7 we have that there exists a scheduler allowing $m[\mathtt{out}\langle c_{L,r}, \tilde{v}\rangle.P]_l \to [\![m[P]_l]\!]_{\Delta}$, and again by Definition 2.7 there exists a scheduler allowing the reduction $m[\mathtt{out}\langle c_{L,r}, \tilde{v}\rangle.P]_l \mid N \mid O \to^* [\![m[P]_l \mid N' \mid O']\!]_{\Delta}$, meaning, by Definition 3.5, $F' \in \hat{\mathcal{F}}_{\mathcal{C}}^{N|O}$ as required.

4. $\alpha = c?\vartheta@k$ the proof is analogous as for $\alpha = c?@k$.

Now we proceed with the restriction.

Let $\mathcal{R} = \{((\nu d)M, (\nu d)N) : M \approx_p^{\mathcal{F}} N\}$ be a relation. We need to prove that it is a probabilistic labelled bisimulation w.r.t. $\mathcal{F}$.

Let us consider $\mathcal{C}$: if $P, Q \in \mathcal{C}$, by definition of $\mathcal{R}$, $P \equiv (\nu \bar{d})\bar{P}$, $Q \equiv (\nu \bar{d})\bar{Q}$ and $\bar{P} \approx_p^{\mathcal{F}} \bar{Q}$. But then $\exists \mathcal{D} \in \mathcal{N}/\approx_p^{\mathcal{F}}$ such that $\mathcal{D} = \{\tilde{P} : (\nu \bar{d})\bar{P} \in \mathcal{C}\}$.

We have to prove that, $\forall F \in \hat{\mathcal{F}}_{\mathcal{C}}^{(\nu d)M}$ $\exists F' \in \hat{\mathcal{F}}_{\mathcal{C}}^{(\nu d)N}$ such that, $\forall \mathcal{C} \in \mathcal{N}/\mathcal{R}$, $\forall \alpha$:

1. $\alpha = \tau$ implies that $Prob_{(\nu d)M}^{F}(\xrightarrow{\tau}, \mathcal{C}) = Prob_{(\nu d)N}^{F'}(\Longrightarrow, \mathcal{C})$.

   Since $\mathtt{Chan}(\tau) = \perp$, by Definition 3.5 $\exists F_1 \in \hat{\mathcal{F}}_{\mathcal{C}}^{M}$ such that $Prob_{(\nu d)M}^{F}(\xrightarrow{\tau}, \mathcal{C}) = Prob_{M}^{F_1}(\xrightarrow{\tau}, \mathcal{D})$ and, since $M \approx_p^{\mathcal{F}} N$ $\exists F_2 \in \hat{\mathcal{F}}_{\mathcal{C}}^{N}$ such that: $Prob_{M}^{F_1}(\xrightarrow{\tau}, \mathcal{D}) = Prob_{N}^{F_2}(\Longrightarrow, \mathcal{D})$.

   Finally we can take $F' \in LSched$ mimicking the executions in the set $Exec_{N}^{F_2}(\Longrightarrow, \mathcal{D})$, when applying the restriction on N. Hence: $Prob_{N}^{F_2}(\Longrightarrow, \mathcal{D}) = Prob_{(\nu d)N}^{F'}(\Longrightarrow, \mathcal{C})$.

   In order to prove that $F' \in \hat{\mathcal{F}}_{\mathcal{C}}^{(\nu d)N}$, we start by the consideration that, by Definition 2.7, for any context there exists a scheduler in $\mathcal{F}_{\mathcal{C}}^{(\nu d)N}$ mimicking the behaviour of N when interacting with the given context. Hence $\exists \bar{F} \in \mathcal{F}_{\mathcal{C}}^{(\nu d)N}$ such that $Exec_{(\nu d)N}^{\bar{F}}$ contains all the reductions corresponding to the executions in $Exec_{(\nu d)N}^{F'}$, meaning, by Definition 3.5, $F' \in \hat{\mathcal{F}}_{\mathcal{C}}^{(\nu d)N}$ as required.

2. $\alpha = c!\tilde{v}@K \lhd R$. Since $d \neq c$, by Definition 3.5 $\exists F_1 \in \hat{\mathcal{F}}_{\mathcal{C}}^{M}$ such that $Prob_{(\nu d)M}^{F}(\xrightarrow{\alpha}, \mathcal{C}) = Prob_{M}^{F_1}(\xrightarrow{\alpha}, \mathcal{D})$, then since $M \approx_p^{\mathcal{F}} N$, $\exists F_2 \in \hat{\mathcal{F}}_{\mathcal{C}}^{N}$ such that $Prob_{M}^{F_1}(\xrightarrow{\alpha}, \mathcal{D}) = Prob_{N}^{F_2}(\overset{\alpha}{\Longrightarrow}, \mathcal{D})$.

   Therefore, since $\mathtt{Chan}(\alpha) \neq d$, $\exists F' \in LSched$ such that:

$$Prob_{N}^{F_2}(\overset{\alpha}{\Longrightarrow}, \mathcal{D}) = Prob_{(\nu d)N}^{F'}(\overset{\alpha}{\Longrightarrow}, \mathcal{C}).$$

Again, we prove that $F' \in \hat{\mathcal{F}}_{\mathcal{C}}^{(\nu d)N}$ as for the previous case.

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

3. $\alpha = c?@k$. Again, since $d \neq c$, by Definition 3.5 $\exists F_1 \in \hat{\mathcal{F}}_{\mathcal{C}}^{M}$ such that $Prob^{F}_{(\nu d)M}(\xrightarrow{\alpha}, \mathcal{C}) = Prob^{F_1}_{M}(\xrightarrow{\alpha}, \mathcal{D})$. Since $M \approx_p^{\mathcal{F}} N$, there exists $F_2 \in \hat{\mathcal{F}}_{\mathcal{C}}^{N}$ such that $Prob^{F_1}_{M}(\xrightarrow{\alpha}, \mathcal{D}) = Prob^{F_2}_{N}(\stackrel{\alpha}{\Longrightarrow}, \mathcal{D})$ or $Prob^{F_1}_{M}(\xrightarrow{\alpha}, \mathcal{D}) = Prob^{F_2}_{N}(\Longrightarrow, \mathcal{D})$. In both cases we can apply rule (Res) to N, since $\mathtt{Chan}(\tau) \neq \mathtt{Chan}(\alpha) \neq d$. Therefore, there exists $F' \in \mathit{LSched}$ such that the required result holds, that is

$$Prob^{F_2}_{N}(\stackrel{\alpha}{\Longrightarrow}, \mathcal{D}) = Prob^{F'}_{(\nu d)N}(\stackrel{\alpha}{\Longrightarrow}, \mathcal{C}) \quad \text{or}$$
$$Prob^{F_2}_{N}(\Longrightarrow, \mathcal{D}) = Prob^{F'}_{(\nu d)N}(\Longrightarrow, \mathcal{C}).$$

In order to prove that $F' \in \hat{\mathcal{F}}_{\mathcal{C}}^{(\nu d)N}$ we proceed as for the previous cases.

4. $\alpha = c?\vartheta@k$. The proof is analogous to the one for $\alpha = c?@k$. □

Proof of Theorem 3.11. In order to prove the completeness we show that the relation $\mathcal{R} = \{(M, N) : M \cong_p^{\mathcal{F}} N\}$ is a probabilistic labelled bisimulation. We have to prove that, $\forall F \in \hat{\mathcal{F}}_{\mathcal{C}}^{M}\ \exists F' \in \hat{\mathcal{F}}_{\mathcal{C}}^{N}$ such that, $\forall \mathcal{C} \in \mathcal{N}/\mathcal{R}, \forall \alpha$:

if $\alpha = \tau$ then $Prob^{F}_{M}(\xrightarrow{\tau}, \mathcal{C}) = Prob^{F'}_{N}(\Longrightarrow, \mathcal{C})$.

By Theorem 3.3 we know that there exists a scheduler $\bar{F} \in \mathit{Sched}$ such that $Prob^{F}_{M}(\xrightarrow{\tau}, \mathcal{C}) = Prob^{\bar{F}}_{M}(\mathcal{C})$, and, by Definition 3.5 we deduce $\bar{F} \in \mathcal{F}_{\mathcal{C}}^{M}$. Since $M \cong_p^{\mathcal{F}} N$, $\exists \bar{F}' \in \mathcal{F}_{\mathcal{C}}^{N}$ such that $Prob^{\bar{F}}_{M}(\mathcal{C}) = Prob^{\bar{F}'}_{N}(\mathcal{C})$. Again by Theorem 3.3 and Definition 3.5, there exists $F' \in \hat{\mathcal{F}}_{\mathcal{C}}^{N}$ such that $Prob^{\bar{F}'}_{N}(\mathcal{C}) = Prob^{F'}_{N}(\Longrightarrow, \mathcal{C} \cup \{\bar{N} \equiv N' \in \mathcal{C}\})$, but since $\cong_p^{\mathcal{F}}$ is closed under structural equivalence, $\forall \bar{N} \equiv N' \in \mathcal{C}, \bar{N} \in \mathcal{C}$, hence:

$$Prob^{F}_{M}(\xrightarrow{\tau}, \mathcal{C}) = Prob^{F'}_{N}(\Longrightarrow, \mathcal{C}).$$

if $\alpha = c!\tilde{v}@K \lhd R$ then $Prob^{F}_{M}(\xrightarrow{\alpha}, \mathcal{C}) = Prob^{F'}_{N}(\stackrel{\alpha}{\Longrightarrow}, \mathcal{C})$.

First we notice that $Prob^{F}_{M}(\xrightarrow{c!\tilde{v}@K \lhd R}, \mathcal{C})$ is either 0 or 1.

If $Prob^{F}_{M}(\xrightarrow{c!\tilde{v}@K \lhd R}, \mathcal{C}) = 0$ we are done, because it will be enough to take any scheduler $F' \in \hat{\mathcal{F}}_{\mathcal{C}}^{N}$ not allowing observable output actions on the channel c, and we get $Prob^{F}_{M}(\xrightarrow{c!\tilde{v}@K \lhd R}, \mathcal{C}) = Prob^{F'}_{N}(\stackrel{c!\tilde{v}@K \lhd R}{\Longrightarrow}, \mathcal{C})$.

If $Prob^{F}_{M}(\xrightarrow{c!\tilde{v}@K \lhd R}, \mathcal{C}) = 1$, then, by Definition 3.5 there exists a scheduler $\bar{F} \in \mathcal{F}_{\mathcal{C}}^{M}$ such that $M \Downarrow_1^{\bar{F}} c@K$, and it means that $\exists \bar{F}' \in \mathcal{F}_{\mathcal{C}}^{N}$ such that $N \Downarrow_1^{\bar{F}'} c@K$, hence $\exists R'$ such that $K \subseteq R'$ and $N \stackrel{c!\tilde{v}@K \lhd R'}{\Longrightarrow}$. Now in order to mimic the effect of the action $c!\tilde{v}@K \lhd R$, we build the context

$$C[\cdot] = \prod_{i=1}^{n}(n_i[\mathtt{in}(c, \tilde{x}_i).[\tilde{x}_i = \tilde{v}]\mathtt{out}\langle \mathtt{f}_{ik_{i,r}}, \tilde{x}_i\rangle]_{k_i} \mid m_i[\mathtt{in}(\mathtt{f}_i, \tilde{y}_i).\mathtt{out}\langle \mathtt{ok}_{ik_{i,r}}, \tilde{y}_i\rangle]_{k_i}),$$

where $R = \{k_1, \ldots, k_n\}$ and $\mathtt{f}_i$ and $\mathtt{ok}_i$ fresh $\forall i \in [1 - n]$.

Since $M \xrightarrow{c!\tilde{v}@K \lhd R}$, then the message is reachable by all nodes n_i, hence, by Definition 2.7, which captures the behaviour of a network when interacting in any context, since $C[M]$ is well-formed, $\exists \bar{F}_1 \in \mathcal{F}_{\mathcal{C}}^{C[M]}$ such that $C[M] \rightarrow^* \bar{M}$, where

$$\bar{M} \equiv M' \mid \prod_{i=1}^{n}(n_i[\mathbf{0}]_{k_i} \mid m_i[\mathtt{out}\langle \mathtt{ok}_{ik_{i,r}}, \tilde{v}_i\rangle]_{k_i}),$$

with $\bar{M} \not\downarrow_{\mathtt{f}_i@R}$ and $\bar{M} \Downarrow_1^{\bar{F}_1} \mathtt{ok}_i@R$, $\forall i \in [1 - n]$.

The absence of the barb on the channels $\mathtt{f}_i$ together with the presence of the barb on the channels $\mathtt{ok}_i$ ensures that all the locations in R have been able to receive the message. Since $C[M] \cong_p^{\mathcal{F}} C[N]$, $\exists \bar{F}_2 \in \mathcal{F}_{\mathcal{C}}^{C[N]}$ such that $Prob^{\bar{F}_1}_{C[M]}(\mathcal{C}') = Prob^{\bar{F}_2}_{C[N]}(\mathcal{C}')$ where $\bar{M} \in \mathcal{C}'$.

Therefore, $C[N] \rightarrow^* \bar{N}$ with $\bar{N} \not\downarrow_{\mathtt{f}@R}$ and $\bar{N} \Downarrow_1^{\bar{F}_2} \mathtt{ok}@R$. The constrains on the barbs allow us to deduce that

$$\bar{N} \equiv N' \mid \prod_{i=1}^{n}(n_i[\mathbf{0}]_{k_i} \mid m_i[\mathtt{out}\langle \mathtt{ok}_{ik_{i,r}}, \tilde{v}_i\rangle]_{k_i})$$

which implies $N \stackrel{c!\tilde{v}@K \lhd R}{\Longrightarrow} N'$, or $N \Longrightarrow N'$ in case (Lose2) has been applied to the output action on the channel c. Since $\bar{M}, \bar{N} \in \mathcal{C}$, then $\bar{M} \cong_p^{\mathcal{F}} \bar{N}$. Since $\cong_p^{\mathcal{F}}$ is contextual, it results $(\nu \mathtt{ok})\bar{M} \cong_p^{\mathcal{F}} (\nu \mathtt{ok})\bar{N}$, from which we can derive that $M' \cong_p^{\mathcal{F}} N'$.

But since $N' \in \mathcal{C}$ and $N \stackrel{c!\tilde{v}@K \lhd R}{\Longrightarrow} N'$, then, by Definition 3.5 $\exists F' \in \hat{\mathcal{F}}_{\mathcal{C}}^{N}$ such that:

$$Prob_N^{F'}\left(\stackrel{c!\tilde{v}@K \lhd R}{\Longrightarrow}, \mathcal{C}\right) = 1 = Prob_M^{F}\left(\stackrel{c!\tilde{v}@K \lhd R}{\Longrightarrow}, \mathcal{C}\right).$$

if $\alpha = c?@k$ then we notice that $Prob_M^F(\xrightarrow{c?\tilde{v}@k}, \mathcal{C})$ is either 0 or 1.

If $Prob_M^F(\xrightarrow{c?@k}, \mathcal{C}) = 0$ we are done, because it will be enough to take any scheduler $F' \in \hat{\mathcal{F}}_{\mathcal{C}}^{N}$ not allowing input actions on the channel c, and we get $Prob_M^F(\xrightarrow{c?@k}, \mathcal{C}) = Prob_N^{F'}(\stackrel{c?@k}{\Longrightarrow}, \mathcal{C})$.

If $Prob_M^F(\xrightarrow{c?@k}, \mathcal{C}) = 1$, because $M \xrightarrow{c?@k} [\![M']\!]_{\Delta}$, by Definition 2.7 there exists at least a context $C[\cdot]$ and $\exists \bar{F} \in \mathcal{F}_{\mathcal{C}}^{C[M]}$ such that $C[M] \to C'[M']$, and by Theorem 3.3 we deduce that:

$$C[\cdot] \equiv (\nu \tilde{d}) m[\mathtt{out}\langle c_{L,r}, \tilde{v}\rangle.P]_l \mid M_1,$$

and

$$C'[\cdot] \equiv (\nu \tilde{d}) m[\bar{c}_{L,r}\langle \tilde{v}\rangle.P]_l \mid M_1',$$

for some m, some tuple $\tilde{d}$ of channel such that $c \notin \tilde{d}$, dome set L of messages, some radius r, some process P, some location l such that $d(l, k) \leq r$ and some (possibly empty) network M_1 and M_1'.

By Definition 2.7, for any context there exists a scheduler in $\mathcal{F}_{\mathcal{C}}^{C[M]}$ allowing m to perform the output when interacting with any context. Hence we can build the following context:

$$C_1[\cdot] = \cdot \mid m[\mathtt{out}\langle c_{L,r}, \tilde{v}\rangle.P]_l \mid m_1[\mathtt{in}(c, \tilde{x}).\mathtt{out}\langle \mathtt{f}_{k,r'}, \tilde{x}\rangle.\mathtt{out}\langle \mathtt{ok}_{k,r'}, \tilde{x}\rangle]_k,$$

in order to mimic the behaviour of the networks, with m static, $\mathtt{f}$ and $\mathtt{ok}$ fresh, $r' > 0$ and $d(l, k) > r' \; \forall l \in \mathbf{Loc}$ s.t. $l \neq k$. There exists a scheduler $\bar{F}_1 \in \mathcal{F}_{\mathcal{C}}^{C_1[M]}$ such that:

$$C_1[M] \to^* M' \mid m[P]_l \mid m_1[\mathtt{out}\langle \mathtt{ok}_{k,r'}, \tilde{v}\rangle]_k \in Exec_{C_1[M]}^{\bar{F}_1},$$

with $M' \mid m[P]_l \mid m[\mathtt{out}\langle \mathtt{ok}_{k,r'}, \tilde{v}\rangle]_k \not\downarrow_{\mathtt{f}@k}$ and

$$M' \mid m[P]_l \mid m[\mathtt{out}\langle \mathtt{ok}_{k,r'}, \tilde{v}\rangle]_k \Downarrow_1^{\bar{F}_1} \mathtt{ok}@k.$$

The reduction sequence above must be matched by a corresponding reduction sequence $C_1[N] \to^* N' \mid m[P]_l \mid m[\mathtt{out}\langle \mathtt{ok}_{k,r'}, \tilde{v}\rangle]_k$, with

$$M' \mid m[P]_l \mid m[\mathtt{out}\langle \mathtt{ok}_{k,r'}, \tilde{v}\rangle]_k \cong_p N' \mid m[P]_l \mid m[\mathtt{out}\langle \mathtt{ok}_{k,r'}, \tilde{v}\rangle]_k,$$
$$N' \mid m[P]_l \mid m[\mathtt{out}\langle \mathtt{ok}_{k,r'}, \tilde{v}\rangle]_k \not\downarrow_{\mathtt{f}@k} \quad \text{and}$$
$$N' \mid m[P]_l \mid m[\mathtt{out}\langle \mathtt{ok}_{k,r'}, \tilde{v}\rangle]_k \Downarrow_1^{\bar{F}_2} \mathtt{ok}@k \quad \text{for some } \bar{F}_2 \in \mathcal{F}_{\mathcal{C}}^{C_1[N]}.$$

This does not ensure that N actually performed the input action, but we can conclude that there exists $F' \in LSched$ and N' such that either $N \stackrel{c?@k}{\Longrightarrow} N'$ or $N \Longrightarrow N'$. Since $M' \mid m[P]_l \mid m[\mathtt{out}\langle \mathtt{ok}_{k,r'}, \tilde{v}\rangle]_k \cong_p N' \mid m[P]_l \mid m[\mathtt{out}\langle \mathtt{ok}_{k,r'}, \tilde{v}\rangle]_k$ and $\cong_p^{\mathcal{F}}$ is preserved by the parallel composition, we can easily derive $M' \cong_p^{\mathcal{F}} N'$ (applying rules for structural equivalence), that means $M', N' \in \mathcal{C}$ and $\exists F' \in LSched$ such that:

$$Prob_M^F(\xrightarrow{c?@k}, \mathcal{C}) = 1 = Prob_N^{F'}(\stackrel{c?@k}{\Longrightarrow}, \mathcal{C}) \quad \text{or}$$
$$Prob_M^F(\xrightarrow{c?@k}, \mathcal{C}) = 1 = Prob_N^{F'}(\Longrightarrow, \mathcal{C}).$$

Now we have only to prove that $F' \in \hat{\mathcal{F}}_{\mathcal{C}}^{N}$, but this follows straightforwardly by Definition 3.5, since $\bar{F}_2 \in \mathcal{F}_{\mathcal{C}}^{C_1[N]}$.
if $\alpha = c?\vartheta@k$ the proof is analogous as for $\alpha = \alpha = c?@k$. □

References

[1] X. Wang, K. Kar, Throughput modelling and fairness issues in CSMA/CA based ad-hoc networks, in: Proc. of the 24th Annual Joint Conf. of the IEEE Computer and Communications Societies, INFOCOM'05, IEEE, 2005, pp. 23–34.
[2] A. Muqattash, M. Krunz, CDMA-based MAC protocol for wireless ad hoc networks, in: Proc. of the 4th ACM International Symposium on Mobile ad hoc Networking & Computing, MobiHoc'03, ACM, 2003, pp. 153–164.
[3] L. Gallina, S. Hamadou, A. Marin, S. Rossi, A probabilistic energy-aware model for mobile ad-hoc networks, in: Proc. of the 18th International Conference on Analytical and Stochastic Modelling Techniques and Applications, ASMTA'11, in: LNCS, vol. 6751, Springer-Verlag, 2011, pp. 316–330.
[4] L. Gallina, S. Rossi, Sender- and receiver-centered interference in wireless ad hoc networks, in: Proc. of IFIP Wireless Days 2010, WD'10, IEEE, 2010.
[5] I. Lanese, D. Sangiorgi, An operational semantics for a calculus for wireless systems, Theoret. Comput. Sci. 411 (19) (2010) 1928–1948.
[6] M. Merro, An observational theory for mobile ad hoc networks, Inf. Comput. 207 (2) (2009) 194–208.
[7] R. Segala, N. Lynch, Probabilistic simulations for probabilistic processes, in: Proc. of the 5th International Conference on Concurrency Theory, CONCUR'94, in: LNCS, vol. 836, Springer-Verlag, 1994, pp. 481–496.

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003

[8] R. Milner, D. Sangiorgi, Barbed bisimulation, in: Proc. of International Colloquium on Automata, Languages and Programming, ICALP'92, in: LNCS, vol. 623, Springer-Verlag, 1992, pp. 685–695.
[9] Ad hoc on-demand distance vector routing protocol, available at http://moment.cs.ucsb.edu/AODV.
[10] Y. Ko, N. Vaidya, Location aided routing (lar) in mobile ad hoc networks, Wirel. Netw. 6 (2000) 307–321.
[11] R. Milner, Communication and Concurrency, Prentice-Hall, 1989.
[12] A. Singh, C. Ramakrishnan, S. Smolka, A process calculus for mobile ad hoc networks, in: Proc. of the 10th International Conference on Coordination Models and Languages, COORDINATION'08, in: LNCS, vol. 5052, Springer-Verlag, 2008, pp. 296–314.
[13] S. Nanz, C. Hankin, A framework for security analysis of mobile wireless networks, Theoret. Comput. Sci. 367 (1) (2006) 203–227.
[14] K.V.S. Prasad, A calculus of broadcasting systems, Sci. Comput. Program. 25 (2–3) (1995) 285–327.
[15] A. Fehnker, R. van Glabbeek, P. Höfner, A. McIver, M. Portmann, W. Tan, A process algebra for wireless mesh networks, in: Programming Languages and Systems, in: LNCS, vol. 7211, Springer, Berlin, Heidelberg, 2012, pp. 295–315.
[16] L. Song, J. Godskesen, Probabilistic mobility models for mobile and wireless networks, in: Theoretical Computer Science, in: IFIP Advances in Information and Communication Technology, vol. 323, Springer, Boston, 2010, pp. 86–100.
[17] J. Goubault-Larrecq, C. Palamidessi, A. Troina, A probabilistic applied pi-calculus, in: Proc. of the 5th Asian Symposium on Programming Languages and Systems, APLAS'07, in: LNCS, vol. 4807/2009, Springer-Verlag, 2007, pp. 175–190.
[18] D. Macedonio, M. Merro, A semantic analysis of wireless network security protocols, in: NASA Formal Methods, in: LNCS, vol. 7226, Springer, Berlin, Heidelberg, 2012, pp. 403–417.
[19] R. Lanotte, M. Merro, Semantic analysis of gossip protocols for wireless sensor networks, in: CONCUR 2011 Concurrency Theory, in: LNCS, vol. 6901, Springer, Berlin, Heidelberg, 2011, pp. 156–170.
[20] A. Cerone, M. Hennessy, Modelling probabilistic wireless networks, in: Formal Techniques for Distributed Systems, in: LNCS, vol. 7273, Springer, Berlin, Heidelberg, 2012, pp. 135–151.
[21] J. Hillston, A Compositional Approach to Performance Modelling, Cambridge University Press, 1996.
[22] M. Bernardo, M. Bravetti, Performance measure sensitive congruences for Markovian process algebras, Theoret. Comput. Sci. 290 (1) (2003) 117–160.
[23] H. Hermanns, Interactive Markov Chains: The Quest for Quantified Quality, in: LNCS, 2428, Springer-Verlag, 2002.
[24] G. Mohimani, F. Ashtiani, A. Javanmard, M. Hamdi, Mobility modeling, spatial traffic distribution, and probability of connectivity for sparse and dense vehicular ad hoc networks, IEEE Trans. Vehicular Technol. 58 (4) (2009).
[25] M. Beccuti, M.D. Pierro, A. Horvàth, A. Horvàth, K. Farkas, A mean field based methodology for modeling mobility in ad hoc networks, in: Proc. of 73rd IEEE Vehicular Technology Conference, VTC Spring, IEEE, Budapest, HU, 2011, pp. 1–5.
[26] M. Bugliesi, L. Gallina, S. Hamadou, A. Marin, S. Rossi, Interference-sensitive preorders for manets, in: Proc. 9th International Conference on Quantitative Evaluation of SysTems, QEST'12, IEEE, 2012, pp. 189–198.
[27] A. Cerone, M. Hennessy, Modelling probabilistic wireless networks, Log. Methods Comput. Sci. 9 (3) (2013).
[28] A.S. Tanenbaum, Computer Networks, Prentice Hall, 2003.
[29] A. Hinton, M. Kwiatkowska, G. Norman, D. Parker, Prism: a tool for automatic verification of probabilistic systems, in: Proc. of the 12th International Conference on Tools and Algorithms for the Construction and Analysis of Systems, TACAS'06, in: LNCS, vol. 3920, Springer-Verlag, 2006, pp. 441–444.
[30] M.Z. Kwiatkowska, G. Norman, D. Parker, Prism 4.0: verification of probabilistic real-time systems, in: G. Gopalakrishnan, S. Qadeer (Eds.), CAV, in: Lecture Notes in Computer Science, vol. 6806, Springer, 2011, pp. 585–591.

Michele Bugliesi (Ph.D. Paris VII) is Professor of Computer Science at the University of Venice "Ca' Foscari" since 2006. Formerly he held various positions in Venice (1998–2006) and Padua (1992–1998), Boston University (1999), ENS Paris (Feb 2000).

He is the author of more than ninety articles in international journals and conferences. He has been the co-recipient of the EATCS best paper award at ETAPS 2013. His current research centres on formal models and static analysis techniques for the automatic verification of security and reliability of distributed and pervasive computing systems. His past research has focused on the semantics of modular extension of declarative languages, and typed theoretical calculi for object-oriented systems.

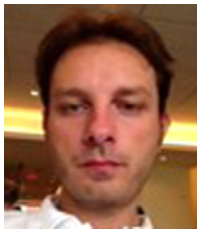
Lucia Gallina received the bachelor and master degrees in Computer Science from the Ca' Foscari University of Venice, Italy, respectively in 2006 and 2009. In April 2013, she got the Ph.D. in Computer Science from the Ca' Foscari University.

Her main research interests concern the study and implementation of formal models for the analysis of mobile ad-hoc networks and sensor networks.

Sardaouna Hamadou is a research fellow in computer science at INRIA Saclay, Ecole Polytechnique. He received his Ph.D. degree in computer science and computer engineering from Ecole Polytechnique de Montreal (University of Montreal). Most of his research activity concerns the foundations of security for mobile, distributed systems, and aims at developing foundational theories that can be applied to present and emerging security problems.

His interests include formal methods, semantics, logic, and in general the foundations of computer science, with main focus on: mathematical modelling and analysis of anonymity, privacy, trust, and routing security in communication systems; theory and design for economic, usability, and incentives analysis of anonymous and route-secure communication systems; probabilistic and quantitative approaches to the analyses of mobile ad hoc networks. He has published more than 15 papers in various areas of computer and network security.

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003



Andrea Marin is an assistant professor of Computer Science at the University Ca' Foscari of Venice since 2011. He received his Ph.D. degree in Computer Science in 2007 from the same university. His main research interests include stochastic modelling of computer and communication systems for performance and reliability analysis, queueing theory, and models with product-form solutions. He has contributed in developing a probabilistic calculus for the formal analysis of wireless ad-hoc networks.

Sabina Rossi received her Ph.D. in Computational Mathematics and Informatics from the University of Padova in 1994. She is Associate Professor of Computer Science at the University Ca' Foscari of Venice since Nov. 2012. Formerly she has been Assistant Professor at Ca' Foscari (2000–2012), visiting professor at Universitè Paris 7 (2007) and research fellow at the Universitè Catholique de Louvain-la-Neuve, Belgium (1997). She is the (co-)author of over 50 technical papers in refereed international journals and conference proceedings. Her current research focuses on the development of formal tools for the analysis and verification based on process algebraic techniques and, specifically, on stochastic process algebras. Sabina Rossi has been an invited speaker at IFIP Theoretical Computer Science TCS 2010 and has been in the program committees of various international conferences and workshops.

Please cite this article in press as: M. Bugliesi, et al., Behavioural equivalences and interference metrics for mobile ad-hoc networks, Performance Evaluation (2013), http://dx.doi.org/10.1016/j.peva.2013.11.003