Telecommunication Systems manuscript No.
(will be inserted by the editor)

Connectivity and Energy-aware Preorders for Mobile Ad-Hoc Networks

Lucia Gallina · Andrea Marin · Sabina Rossi

Received: date / Accepted: date

Abstract Network connectivity and energy conservation are two major goals in mobile ad-hoc networks (MANETs). In this paper we propose a probabilistic, energy-aware, broadcast calculus for the analysis of both such aspects of MANETs. We first present a probabilistic behavioural congruence together with a co-inductive proof technique based on the notion of bisimulation. Then we define an energy-aware preorder over networks. The behavioural congruence allows us to verify whether two networks exhibit the same (probabilistic) connectivity behaviour, while the preorder makes it possible to evaluate the energy consumption of different, but behaviourally equivalent, networks. In practice, the quantitative evaluation of the models is carried out by resorting to the statistical model checking implemented in the PRISM tool, i.e., a simulation of the probabilistic model. We consider two case studies: first we evaluate the performance of the Location Aided Routing (LAR) protocol, then we compare the energy efficiency of the Go-Back-N protocol with that of the Stop-And-Wait in a network with mobility.

Keywords Manets · Process Algebras · Energy Conservation · Performance Evaluation · Simulation

1 Introduction

Mobile ad-hoc networks (MANETs) consist of mobile devices connected by wireless links and communicating with each other without any pre-existing infrastructure. Nodes are free to move arbitrarily in any direction, and therefore their links to other nodes may change frequently. Moreover, since mobile devices are often dependent on battery power, it is important to minimize their energy consumption. As a consequence, one of the major issues of current communication protocols is that of providing a full connectivity among the network devices while maintaining good performances both in terms of throughput and of energy conservation (see, e.g., [1–6]). For larger networks in which some of/all the nodes are aware of their relative or absolute geographical position, e.g., thanks to a Global Positioning System device (GPS), the routing protocols may exploit this information in order to improve the efficiency of packet delivery by controlling the flooding process (see, e.g., [7,8]).

Drawing on earlier work on the subject [9–12], in this paper we present a calculus for the analysis of network connectivity and the evaluation of energy consumption in mobile ad-hoc networks.

The definition of a general framework for both qualitative (connectivity) and quantitative (power consumption and throughput) analysis is a challenging topic of research. Indeed, general purpose formalisms for concurrency (e.g., Petri nets) do not deal with the mobility of the devices in a natural way, and hence they do not allow for a modular and hierarchical description of mobile systems. In [13] we presented a calculus with non-atomic output and input actions to capture the presence of interferences caused by the simultaneous transmission of two (or more) nodes. The calculus of [13] is targeted at the evaluation of the level of interference in mobile ad hoc networks, while no quantitative assessment of energy consumption is considered. Here we present a calculus, named Probabilistic EBUM, for formally reasoning about Energy-aware Broadcast, Uni-

Lucia Gallina · Andrea Marin · Sabina Rossi
DAIS, Università Ca' Foscari Venezia, via Torino 155, 30172 Mestre Venezia, Italy
Tel.: +39 041 2348411 Fax: +39 041 2348419
E-mail: {lgallina,marin,srossi}@dais.unive.it

cast and Multicast communications of mobile ad-hoc networks. This is an extension of the EBUM calculus presented in [10,14,15] where probability distributions are used to describe the movements of nodes. As in previous works [10,13], our calculus allows us to represent the system devices as syntactical terms, named nodes, associated with labels, named locations, allowing us to identify the transmission area of each communication. Differently from [10], nodes may move inside the network according to a fixed probability distribution. Wireless synchronizations are non-deterministic, and modeled by sequential processes inside the nodes. The calculus allows us to model broadcast, unicast and multicast communications limited to the transmission cell of the sender. The idea of using location-based destination is motivated by the need of efficiently modelling large networks with location-based routing (see, e.g., [7, 8]), and of comparing their efficiency with respect to standard routing algorithms based on flooding. Nevertheless, the routing based on the knowledge of nodes destination addresses (but not their physical locations) can still be implemented in our calculus by specifying the intended recipients' addresses as part of the message content. This reflects the actual implementation of wireless protocols in which messages are broadcast and then filtered by the recipient devices according to the (MAC) address specified in the header of the packet. Another important feature of the calculus is the fact that nodes may control their transmission power by modifying the communication transmission radius.

The semantics of our calculus is expressed in terms of both probabilistic and non-deterministic transitions. Schedulers are used to resolve the non-deterministic choice among different probability distributions over target states. This leads to a purely probabilistic model that can be studied by resorting to both exact and statistical approaches. In fact, for most of practical applications the cardinality of the model state space is huge enough to make the application of standard probabilistic analysis techniques prohibitive from the computational point of view. We show that our calculus can be implemented within the PRISM model checker [16] and hence the discrete-event simulator built into PRISM can be used to perform a statistical model checking.

In this paper we define a probabilistic behavioural congruence in the style of [17] to equate networks exhibiting the same probabilistic connectivity behaviour. As in [15,14], and in contrast to [12], the notion of observability is relative to the nodes listening at specific locations in the network, so as to allow a fine grained analysis of connectivity at different areas within a network. We give a coinductive characterisation of behavioural congruence based on a labelled transition semantics. This is a bisimulation-based proof technique in the form of a probabilistic labelled bisimilarity which is shown to coincide with the behavioural equivalence. We also introduce energy-aware preorders over networks to measure the relative energy cost of different, but behaviourally equivalent, networks. We present two case-studies. The first one consists in modelling the Location Aided Routing (LAR) protocol [7]: we study how the performances of this protocol vary depending on the characteristics of the specific network, e.g., node density, topology changes and power capacity of the devices. In the second case-study we compare the performances, in terms of energy consumption, of an aggressive protocol for reliable communications (Go-Back-n) and a slower protocol (Stop&Wait).

This paper is an extended and improved version of [9]. The main novelties concern the extension of the calculus through the channel restriction operator (νc) over networks that is useful to specialise the verification method to some specific class of contexts. Moreover, we define a new equivalence relation that is parametric to a restricted set of executions for a given network: our new definition of *probabilistic barbed congruence* allows us to study the performances of networks focusing the attention only on specific restricted behaviours, abstracting out all the executions that are unrealistic or that are simply non interesting for the aims of the analysis. We also define the labelled semantics which is proved to coincide with the probabilistic behavioural congruence. This provides the basis for powerful, both inductive and co-inductive, proof techniques. Finally, the analysis of the LAR protocol using our Probabilistic EBUM calculus is totally new. For this case study we perform a quantitative analysis based on the discrete-event simulation that is available in the PRISM model checker.

1.1 Related work

Various probabilistic algebraic calculi have been developed to model mobile ad-hoc and sensor networks.

Song and Godskesen [18] propose a probabilistic broadcast calculus for mobile and wireless networks with unreliable connections. The main feature of this calculus is the presence of a *probabilistic mobility function* to model the mobility of nodes. Recently, in [19] the same authors propose a new version of their calculus built upon a *stochastic mobility function* to model the stochastic changes of connectivity. As in our works [14, 9,10] broadcast actions are associated with the locations of the intended recipients of the message. However, differently from our calculus, in [19] any notion of transmission radius is introduced and any performance analysis is considered.

Palamidessi et al. in [20] define the Probabilistic Applied π-calculus: this is a probabilistic extension of Applied π-calculus [21], where both non-deterministic and probabilistic choices are modelled. The authors define both a static equivalence, and an observational congruence based on the notion of probabilistic barb, which describes the probability, for a given system, to perform a certain observable action. As in our calculus, in order to solve the non-determinism, schedulers (also called polices, or adversaries) have been introduced. They are modelled as functions mapping states into probability distributions. Differently from our work, their semantic is not parametrized over restricted sets of schedulers.

Merro et al. introduce aTCWS (applied Timed Calculus for Wireless Systems) [22]: a timed broadcasting process algebra for the analysis of security properties of wireless networks with fixed nodes, all using the same transmission radius for their communications. The connectivity of the network is expressed by associating with each node a tag containing the list of all its neighbours. The timed model adopted by this calculus is known as the *fictitious clock* approach, and it is based on clock synchronization of nodes. A probabilistic version of TCWS has been introduced in [23]. The main feature of this calculus is the presence of a *simulation up to probability* which allows one to compare networks which exhibit the same behaviour up to a certain probability. The main limitations of such calculus are the absence of mobility and of multiple frequencies.

In [24] Hennessy and Cerone propose a calculus to model the high-level behaviour of Wireless Systems (i.e., MAC-layer protocols). As our calculus, the one presented in [24] models both probabilistic and non-deterministic behaviours. Differently from our model, it does not rely on any notion of distance or transmission radius but represents the topology of the network as an indirect graph where each edge denotes a link between two nodes. More precisely, it presupposes that all nodes use the same transmission radius to communicate, an assumption that is not realistic for MANETs, which include different kinds of devices, with different physical structure and power resources.

De Nicola et al. introduce StoKlaim [25]: a stochastic process algebra, whose underlying processes are Continuous Time Markov Chains, allowing one to describe random phenomena regarding mobile wireless networks.

As far as performance evaluation is concerned, Hillston et al. introduce the process algebra PEPA [26] which has been designed for reasoning about systems composed of concurrent components which co-operate each other and share resources. The authors also provide a tool, the PEPA Workbench [27], which allows a practical use of this process algebra in many applications concerning software architecture and communication protocols.

Bernardo et al. introduce EMPA$_{gr}$ [28], an extended Markovian process algebra including probabilities, priorities and exponentially distributed durations. The peculiarity of this calculus is the possibility of modelling both exponentially timed and immediate actions, whose selection is controlled by associating priority levels.

Other performance modelling approaches are based on Petri Nets and queueing networks but they fall short of accounting for node mobility while maintaining a good accuracy in the protocol specification [29–31].

Many recent works investigate the problem of measuring the energy consumption for specific communication protocols for wireless networks. For instance, in [2] the authors define a Markov Reward process [32] for the analysis of some protocols for pairwise node communications. A steady-state quantitative analysis is then derived and hence the average performance indices computed. In [33] Bernardo et al. present a methodology to predict the impact of the power management techniques on a system functionality and performance. In [1] the authors define a set of metrics for measuring the energy consumption, present various simulations and show how protocols can be modified to improve the efficiency. With respect to those works, the model proposed here aims at providing a common framework for both qualitative and quantitative analyses.

Concerning the problem of routing in mobile ad-hoc networks, several different solutions have been proposed. Usually, routing protocols are classified in *proactive* and *reactive*. While proactive protocols continually exchange routing information about all the nodes, (see, e.g., DSDV [34] and WRP [35]), the reactive protocols update the routing table of each node only on-demand (see, e.g., the AODV [36], TORA [37] and DSR [38]). Although proactive routing reduces the latency in sending out packets, due to the continuous up-to-date of the routing tables, reactive routing are more efficient in terms of resource usage, since they update the route tables only on-demand. When dealing with mobile ad-hoc networks the most common strategy is to use hybrid protocols, where both the proactive and the reactive approach coexist in order to provide a good trade-off between latency and overhead.

1.2 Plan of the paper

Section 2 introduces the Probabilistic EBUM calculus and its semantics. In Section 3 we present the labelled transition semantics and define a labelled bisimilarity which is proved to coincide with the behavioural congruence of the unlabeled semantics. Section 4 shows

how to exploit the labelled semantics for estimating the energy consumption of mobile ad-hoc networks and contrasting the average energy cost of networks exhibiting the same connectivity behaviour. Section 5 analyses the LAR protocol, comparing it with the simple flooding algorithm usually adopted in reactive routing. Section 6 carries out a quantitative and qualitative comparison of the Stop&Wait and Go-Back-N protocols under a specific scenario. Section 7 concludes the paper.

2 A Probabilistic Calculus for MANETs

We introduce the Probabilistic EBUM calculus, an extension of EBUM (a calculus for Energy-aware Broadcast, Unicast, Multicast communications of mobile ad-hoc networks) [15]. The calculus has been designed for modeling mobile ad-hoc networks as a collection of mobile devices, running in parallel, and using channels to broadcast messages. It supports both multicast and unicast communications. Moreover, it allows us to model the capability for a node to adjust its transmission range and then to control the transmission energy.

2.1 Syntax

Hereafter, we use letters c and d for *channels*; m and n for *nodes*; r for *transmission radii*; l, k and h for *locations*; x, y and z for *variables*. *Closed values* include nodes, locations, transmission radii and any basic value (e.g., booleans, integers, ...), while *values* include also variables. Moreover, letters u and v are used for closed values and w for (open) values. We write $\tilde{v}$, $\tilde{w}$ for tuples of values and Loc for the set of all locations.

The syntax of our calculus is shown in Table 1. Networks are collections of nodes (or devices) running in parallel and using channels to broadcasting messages. We denote by $\mathbf{0}$ the empty network and by $M_1|M_2$ the parallel composition of two networks. We will write $\prod_{i\in I} M_i$ to denote the parallel composition of the networks M_i, for $i \in I$. The syntactical term $n[P]_l$ denotes a node n, located at the physical location l, and executing the process P. The channel c is bound with scope M in $(\nu c)M$. Note that in our calculus channels cannot be neither transmitted nor dynamically created and thus $(\nu c)M$ simply plays the role of a CCS-style hiding operator. We write $fc(M)$ for the set of channels which are not bound in M and denote by $\mathcal{N}$ the set of all networks.

Processes are sequential and live within the nodes. The inactive process is denoted by $\mathbf{0}$. The input process $c(\tilde{x}).P$ can receive a tuple $\tilde{w}$ of (closed) values through channel c and then continue as P with $\tilde{x}$ substituted by $\tilde{w}$ (where $|\tilde{x}| = |\tilde{w}|$, and $|\cdot|$ denotes the length of the tuple), i.e., as $P\{\tilde{w}/\tilde{x}\}$. The variables $\tilde{x}$ in $c(\tilde{x}).P$ are said to be bound in P. The output process $\bar{c}_{L,r}\langle\tilde{w}\rangle.P$ can send a tuple of (closed) values $\tilde{w}$ through channel c and then continue as P. The tag L denotes the set of locations of the intended recipients: $L = Loc$ represents a broadcast transmission, while a finite set of locations L denotes a multicast communication (unicast if L is a singleton). This allows us to model the behaviour of location-aware mobile networks where messages can be efficiently routed by specifying the final destination of the recipients by means of their physical address. Notice that in the real implementations of the transmission protocols the destination addresses are included the headers of the packets. The tag r denotes the transmission radius of the sender and is decided by the process running inside the transmitter node. We assume that the transmission radius of a communication cannot exceed the maximum transmission radius associated with the sending node. In a process term, tags L and r associated with an output action on a channel c may be variables, but they must be instantiated when the output prefix is ready to fire. Process $[w_1 = w_2]P, Q$ behaves as P if $w_1 = w_2$, and as Q otherwise. We write $A\langle\tilde{w}\rangle$ to denote a 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 appear free in P. We equate processes up to α-conversion and assume that there are no free variables in a network. We write c_l for $c_{\{l\}}$, $\bar{c}_{L,r}\langle\tilde{w}\rangle$ for $\bar{c}_{L,r}\langle\tilde{w}\rangle.\mathbf{0}$, $\mathbf{0}$ for $n[\mathbf{0}]_l$ and $[w_1 = w_2]P$ for $[w_1 = w_2]P, \mathbf{0}$.

Nodes cannot be added or deleted, and move autonomously. The network connectivity is expressed in terms of node locations and transmission radius: a message broadcast by a node is received only by the nodes lying in the transmission area of the sender. Let $d(\cdot,\cdot)$ be a function which returns the distance between two locations (it can be the Euclidean distance or a more complex function dealing with potential obstacles).

Each node n is characterized by a pair $< r_n, \mathbf{J}^n >$: 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, $\sum_{k\in Loc} \mathbf{J}^n_{lk} = 1$ for all locations $l \in Loc$ and nodes n. Static nodes are associated with the identity Markov chain such that $\mathbf{J}^n_{ll} = 1$ for all $l \in Loc$ and $\mathbf{J}^n_{lk} = 0$ for all $k \neq l$. We denote by μ^n_l the probability distribution associated with node n located at l, that is, the function over Loc such that $\mu^n_l(k) = \mathbf{J}^n_{lk}$, for all $k \in Loc$[1].

[1] Notice that $\mathbf{J}^n$ is a matrix, while μ^n_l is a function.

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

Table 1: Syntax

2.2 Probabilistic network behaviour

Consider a network M with a node n at location l. We write $M\{n : l'/l\}$ to denote the network obtained by substituting l by l' in n and by $[\![M]\!]_{\mu_l^n}$ the probability distribution over the set of networks induced by μ_l^n and defined as follows: for all networks M',

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

More precisely, $[\![M]\!]_{\mu_l^n}(M')$ denotes the probability that the network M evolves to M' because of the movement of n located at l. We say that M' is in the support of $[\![M]\!]_{\mu_l^n}$, denoted $M' \in spt([\![M]\!]_{\mu_l^n})$, if $[\![M]\!]_{\mu_l^n}(M') \neq 0$. Let $[\![M]\!]_{\Delta}$ be the Dirac distribution on the network M such that $[\![M]\!]_{\Delta}(M) = 1$ and $[\![M]\!]_{\Delta}(M') = 0$ for all M' such that $M' \neq M$. Finally, let θ range over $\{\mu_l^n \mid n \text{ is a node and } l \in Loc\} \cup \{\Delta\}$.

Example 1 (Probability distributions) Consider the network defined as

$$M = n_1[\bar{c}_{L,r_1}\langle\tilde{v}_1\rangle.P_1]_{l_1} \mid n_2[\bar{c}_{L,r_2}\langle\tilde{v}_2\rangle.P_2]_{l_2} \mid m[c(\tilde{x}).P_3]_k$$

with two mobile nodes, n_1 and n_2, communicating with a static node m. The nodes n_1 and n_2 move back and forth between the two locations l_1 and l_2 according to the probability distribution defined by the Markov chain with the following transition matrix

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

where $0 < p, q < 1$. The probability distribution of the network induced by the movement of n_1 is

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

Similarly for n_2 we have

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

while for the static receiver m we have

$$[\![M]\!]_{\mu_k^m}(M') \;=\; \begin{cases} 1 & \text{if } M' = M \\ 0 & \text{otherwise} \end{cases}$$

i.e., $[\![M]\!]_{\mu_k^m} = [\![M]\!]_{\Delta}$. □

2.3 Probabilistic reduction semantics

The dynamics of the calculus is expressed in terms of the *probabilistic reduction relation* over networks ($\rightarrow$), described in Table 3. As usual, 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 $M \rightarrow [\![M']\!]_{\theta}$ meaning that a network M evolves to M' according to the probability distribution $[\![M']\!]_{\theta}$.

Rule (R-Bcast) describes the evolution of a network with a sender node n transmitting of a tuple of messages $\tilde{v}$ to the set of locations L through channel c with transmission radius r. Observe that nodes communicate using radio frequencies and broadcasting transmissions (monopolizing channels is not permitted). However, modern routing protocols for manets support multicasting communications allowing nodes to communicate with a specific group of nodes, and this is the reason why we decided to label each output action a set of target locations L. The cardinality of this set indicates the kind of communication that is used: if $L = Loc$ then the recipients set is the whole network and this denotes a broadcast transmission, while if L is a finite set (resp., a singleton) then a multicast (resp., a unicast) communication is performed. Observe that L does not play a role in the (R-Bcast) rule, as messages are broadcast and received by any active receiver in the transmission range. On the other hand, we will use L to fine-tune the notion of observation in the definition of the behavioural semantics. Moreover, since the output is a non-blocking action, the index set I could be empty, i.e., rule (R-Bcast) could be applied even if no nodes are ready to receive the transmission. A radius

$n[\mathbf{0}]_l \equiv \mathbf{0}$	(Struct Zero)
$n[[v = v]P, Q]_l \equiv n[P]_l$	(Struct Then)
$n[[v_1 = v_2]P, Q]_l \equiv n[Q]_l \quad v_1 \neq v_2$	(Struct Else)
$n[A\langle\tilde{v}\rangle]_l \equiv n[P\{\tilde{v}/\tilde{x}\}]_l$ if $A(\tilde{x}) \stackrel{\text{def}}{=} P \wedge \lvert\tilde{x}\rvert = \lvert\tilde{v}\rvert$	(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 2: Structural Congruence

$$\text{(R-Bcast)}\ \frac{}{n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid \prod_{i\in I} n_i[c(\tilde{x}_i).P_i]_{l_i} \rightarrow [\![n[P]_l \mid \prod_{i\in I} n_i[P_i\{\tilde{v}/\tilde{x}_i\}]_{l_i}]\!]_\Delta}$$

where $0 < r \leq r_n$, $\forall i \in I. d(l, l_i) \leq r$, $r_i > 0$ and $|\tilde{x}_i| = |\tilde{v}|$

$$\text{(R-Move)}\ \frac{}{n[P]_l \rightarrow [\![n[P]_l]\!]_{\mu_l^n}} \qquad \text{(R-Par)}\ \frac{M \rightarrow [\![M']\!]_\theta}{M \mid N \rightarrow [\![M' \mid N]\!]_\theta}$$

$$\text{(R-Res)}\ \frac{M \rightarrow [\![M']\!]_\theta}{(\nu\tilde{c})M \rightarrow [\![(\nu\tilde{c})M']\!]_\theta} \qquad \text{(R-Struct)}\ \frac{N \equiv M \ M \rightarrow [\![M']\!]_\theta \ M' \equiv N'}{N \rightarrow [\![N']\!]_\theta}$$

Table 3: Reduction Semantics

r is also associated with any output action, indicating the transmission radius required for that communication which may depend on the energy consumption strategy adopted by the surrounding protocol.

Rule (R-Move) describes node movements within the network. A node n located at l and performing a move action will reach a location with a probability expressed by the distribution μ_l^n that depends on the Markov chain $\mathbf{J}^n$ associated with n. In our model movements are atomic actions. Moreover, due to the interleaving nature of the calculus, only one node can move at each reduction but this does not mean that only one node can move at a time. Indeed, as usual in interleaving semantics, concurrent events are represented by sequentiality and non-determinism. Rules (R-Par), (R-Res) and (R-Struct) are standard.

For a network M, we write $M \rightarrow_\theta N$ when $M \rightarrow [\![M']\!]_\theta$ and N is in the support of $[\![M']\!]_\theta$. An execution for M is a (possibly infinite) sequence of steps

$$M \rightarrow_{\theta_1} M_1 \rightarrow_{\theta_2} M_2 \cdots .$$

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 $M \rightarrow_{\theta_1} M_1 ... \rightarrow_{\theta_j} M_j$ of length j of the execution

$$e = M \rightarrow_{\theta_1} M_1 \cdots \rightarrow_{\theta_j} M_j \rightarrow_{\theta_{j+1}} M_{j+1} \cdots ,$$

and $e\uparrow$ for the set of e' such that e is a prefix of e'. We denote by $\rightarrow^*$ the transitive and reflexive closure of $\rightarrow$.

2.4 Behavioural semantics

We formalize the behavioural semantics for our calculus in terms of a notion *barb*, that provides the basic unit of observation [17]. As in other calculi for wireless communications, the definition of barb is naturally expressed in terms of message transmission. However, the technical development in this paper is more involved, as our calculus presents both non-deterministic and probabilistic aspects, where the non-deterministic choices are among the possible probability distributions that a network may follow and arise from the possibility for nodes to perform movements according to the associated discrete time Markov chain.

We denote by $behave(M) = \{[\![M']\!]_\theta \mid M \rightarrow [\![M']\!]_\theta\}$ 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 [39]).

Definition 1 (Scheduler) A *scheduler* is a total function F assigning to a finite execution e a distribution $[\![N]\!]_\theta \in behave(last(e))$.

Let $Sched$ be the set of all schedulers. Given a network M and a scheduler F, we define the set of executions starting from M and driven by F as:

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

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

$$P_M^F(e) = [\![M'_1]\!]_{\theta_1}(M_1) \cdot ... \cdot [\![M'_k]\!]_{\theta_k}(M_k)$$

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, $\sigma Field_M^F$ is the smallest sigma field on $Exec_M^F$ that contains the basic cylinders $e\uparrow$, where $e \in Exec_M^F$. The probability measure $Prob_M^F$ is the unique measure on $\sigma Field_M^F$ such that $Prob_M^F(e\uparrow) = P_M^F(e)$. Given a measurable set of networks H, we denote by $Exec_M^F(H)$ the set of executions starting from M and crossing a state in H. Formally, $Exec_M^F(H) = \{e \in Exec_M^F \mid last(e^j) \in H$ for some $j\}$. We denote the probability for a network M to evolve into a network H, according to the policy given by F, as $Prob_M^F(H) = Prob_M^F(Exec_M^F(H))$.

The notion of barb introduced below denotes an observable transmission with a certain probability according to a fixed scheduler. In our definition, a transmission is observable only if at least one location in the set of the target locations is able to receive the message.

Definition 2 (Barb) Let $M \equiv (\nu \tilde{d})(n[\bar{c}_{L,r}\langle \tilde{v} \rangle.P]_l | M')$, with $c \notin \tilde{d}$. We say that M has a barb on a channel c at locations $K(\neq \emptyset)$, denoted $M \downarrow_{c@K}$, if $\exists K \subseteq L$ such that $d(l,k) \leqslant r$ for all $k \in K$.

Definition 3 (Probabilistic Barb) A network M has a *probabilistic barb* with probability p on a channel c to the set K of locations, according to the scheduler F, written $M \Downarrow_p^F c@K$, if $Prob_M^F(\{N | N \downarrow_{c@K}\}) = p$.

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

In the following, we introduce a probabilistic behavioural congruence, in the style of [20], which is parametric to a restricted set of schedulers.

Schedulers constitute an essential feature for modeling communication protocols as they provide freedom in modelling implementation and incomplete knowledge of a system. However, many schedulers could be in fact unrealistic. Consider for example schedulers giving priority to communication actions over movements of the nodes. Such schedulers cancel the consequence that nodes mobility has on the network behaviour, since no movements can be performed during the execution.

Therefore our aim is the definition of a relation allowing us to compare networks with respect to a given restricted set of schedulers.

In order to define a congruence relation among networks, we have to select a set of schedulers guaranteeing that, for each behaviour a network can exhibit, the same behaviour can be exhibited by the network in the presence of any possible context. Hereafter, a context is 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 allows us to select the set of schedulers preserving the contextuality, once we have fixed the particular behaviour we want to capture.

Definition 4 Given a scheduler $F \in Sched$, we denote by $F_{\mathcal{C}}$ the set of schedulers F' such that $\forall M_0$, $\forall e \in Exec_{M_0}^F$ of the form

$e = M_0 \to_{\theta_1} M_1 \to_{\theta_2} M_2 ... \to_{\theta_h} M_h$,

$\forall$ context $C_0[\cdot]$ and $\forall e' \in Exec_{C_0[O_0]}^{F'}$ with $M_0 \equiv O_0$ of the form

$e' = C_0[O_0] \to_{\theta'_1} C_1[O_1] \to_{\theta'_2} C_2[O_2] ... \to_{\theta'_k} C_k[O_k]$,

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) $\forall j \in [1-k]$, $\theta'_j = \theta_{f(j)}$ if $M_{f(j-1)} \to_{\theta_{f(j)}} M_{f(j)}$.

Given a subset $\mathcal{F} \in Sched$ of schedulers, then we define $\mathcal{F}_{\mathcal{C}} = \bigcup_{F \in \mathcal{F}} F_{\mathcal{C}}$.

Example 2 Let $M_0 \equiv m[\bar{c}_{L,r}\langle v \rangle.P]_l$ and $F \in Sched$ such that

$$M_0 \to_\Delta M_1 \in Exec_M^F,$$

with $M_1 \equiv m[P]_l$.

First notice that $F \in F_{\mathcal{C}}$, since we can take the empty context $C[\cdot] \equiv [\cdot]$ and the identity function f such that $f(i) = i$ for all $i \in [0-1]$. In this case $C[M_i] \equiv M_i$ for $i \in \{0,1\}$ and the property of Definition 4 is satisfied.

Let $N_0 \equiv n[c(x).Q]_k$ such that $d(l,k) \leq r$. All the schedulers allowing M_0 and N_0 to interact are in $F_{\mathcal{C}}$. Indeed, consider $F_1 \in \mathit{Sched}$ such that, according to rule (R-Bcast),

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

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

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

Both F_1 and F_2 belong to $F_{\mathcal{C}}$.

Now consider again the network N_0. Let

$$e' = n[c(x).Q]_k \rightarrow_{\mu^n_k} n[c(x).Q]_{k'} \notin Exec^{F}_{N_0},$$

then $\forall \bar{F} \in \mathit{Sched}$ such that $e' \in Exec^{\bar{F}}_{N_0}$, $\bar{F} \notin F_{\mathcal{C}}$ since $\bar{F}$ does not satisfy the conditions of Definition 4. □

We are now in position to introduce our equivalence relation.

Definition 5 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 $F \in \mathcal{F}_{\mathcal{C}}$ implies that there exists $F' \in \mathcal{F}_{\mathcal{C}}$ 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 $F \in \mathcal{F}_{\mathcal{C}}$, there exists $F' \in \mathcal{F}_{\mathcal{C}}$ 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 $\mathcal{C}[\cdot]$, it holds that $\mathcal{C}[M]\,\mathcal{R}\,\mathcal{C}[N]$.

The probabilistic behavioural congruence with respect to a restricted set $\mathcal{F}$ of schedulers is defined as the largest relation as follows.

Definition 6 (Probabilistic Behavioural Congruence) Given a set $\mathcal{F}$ of schedulers, the *probabilistic behavioural 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.

Two networks are related by $\cong^{\mathcal{F}}_p$ if they exhibit the same probabilistic behaviour (communications) relative to the corresponding sets of intended recipients. In the next section we develop a bisimulation-based proof technique for $\cong^{\mathcal{F}}_p$. It provides an efficient method to check whether two networks are related by $\cong^{\mathcal{F}}_p$.

3 A Co-Inductive Proof Technique

Proving the relation $\cong^{\mathcal{F}}_p$ may be a hard task. In this section we develop a co-inductive proof technique that allows for an algorithmic decision of $\cong^{\mathcal{F}}_p$.

3.1 Labelled Transition Semantics

We define a *labelled transition semantics* (LTS, for short) 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:

$$\eta ::= c\tilde{v} \mid \bar{c}_{L,r}\tilde{v}.$$

Rules for processes are standard and consist of deterministic transitions only. Table 5 depicts the LTS rules for networks. Transitions are of the form $M \xrightarrow{\gamma} [\![M']\!]_\theta$, where M is a network and $[\![M']\!]_\theta$ is a distribution over networks. Node mobility is expressed in terms of probability distributions. The label γ is as follows:

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

Rule (Snd) describes the behaviour of a node sending a tuple $\tilde{v}$ via channel c to a specific set L of locations with transmission radius r (this is represented by the transistion label $c_L!\tilde{v}[l,r]$), while rule (Rcv) models the reception of $\tilde{v}$ by a node n at l via channel c (represented by the transmission label $c?\tilde{v}@l$).

Broadcasting is modeled by rule (Bcast): messages are received by all the nodes lying within the transmission cell of the sender, independently from the set of intended receivers L.

Rule (Obs) deals with observability: a transmission may be detected (and hence *observed*) by any recipient within the transmission cell of the sender and lying in one of the locations in L. The label $c!\tilde{v}@K \triangleleft R$ denotes the transmission of the tuple $\tilde{v}$ of messages via c: the set R contains all the locations receiving the message, while its subset K contains only the locations where the transmission is observed.

Rule (Lose) models message loss. We use τ-transitions to denote non-observable actions.

Rule (Move) models the movement of a mobile node n from a location l to a location k according to the probability distribution μ^n_l, which is specified by the Markov chain $\mathbf{J}^n$ statically associated with n.

Rule (Res) deals with the standard channel restriction, where $\mathtt{Chan}(\gamma) = c$ if γ is of the form $c?\tilde{v}@l$ or $c_L!\tilde{v}[l,r]$ or $c!\tilde{v}@K \triangleleft R$, and $\mathtt{Chan}(\tau) = \bot$.

Finally, rule (Par) is standard.

3.2 Reduction vs. labelled transition semantics

In this section we prove that the labelled transition semantics coincides with the reduction semantics given in the previous section.

$$\text{(Output)}\ \frac{-}{\bar{c}_{L,r}\langle\tilde{v}\rangle.P \xrightarrow{\bar{c}_{L,r}\tilde{v}} P} \qquad \text{(Input)}\ \frac{-}{c(\tilde{x}).P \xrightarrow{c\tilde{v}} P\{\tilde{v}/\tilde{x}\}}$$

$$\text{(Then)}\ \frac{P \xrightarrow{\eta} P'}{[\tilde{v}\ =\ \tilde{v}]P, Q \xrightarrow{\eta} P'} \qquad \text{(Else)}\ \frac{Q \xrightarrow{\eta} Q' \quad \tilde{v}_1 \neq \tilde{v}_2}{[\tilde{v}_1\ =\ \tilde{v}_2]P, Q \xrightarrow{\eta} Q'}$$

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

Table 4: LTS rules for Processes

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

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

$$\text{(Obs)}\ \frac{M \xrightarrow{c_L!\tilde{v}[l,r]} [\![M']\!]_\Delta \quad R \subseteq \{l' \in Loc:\ d(l,l') \leq r\} \quad K = R \cap L,\ K \neq \emptyset}{M \xrightarrow{c!\tilde{v}@K \triangleleft R} [\![M']\!]_\Delta}$$

$$\text{(Lose)}\ \frac{M \xrightarrow{c_L!\tilde{v}[l,r]} [\![M']\!]_\Delta}{M \xrightarrow{\tau} [\![M']\!]_\Delta} \qquad \text{(Move)}\ \frac{}{n[P]_l \xrightarrow{\tau} [\![n[P]_l]\!]_{\mu_l^n}}$$

$$\text{(Par)}\ \frac{M \xrightarrow{\gamma} [\![M']\!]_\theta}{\begin{array}{c} M|N \xrightarrow{\gamma} [\![M'|N]\!]_\theta \\ N|M \xrightarrow{\gamma} [\![N|M']\!]_\theta \end{array}} \qquad \text{(Res)}\ \frac{M \xrightarrow{\gamma} [\![M']\!]_\theta \quad \texttt{Chan}(\gamma) \neq c}{(\nu c)M \xrightarrow{\gamma} [\![(\nu c)M']\!]_\theta}$$

Table 5: LTS rules for Networks

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 1 *Let M be a network.*

1. *If $M \xrightarrow{c?\tilde{v}@l} [\![M']\!]_\Delta$, 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*

$$M \equiv (\nu\tilde{d})(n[c(\tilde{x}).P]_l | M_1)$$

and

$$M' \equiv (\nu\tilde{d})(n[P\{\tilde{v}/\tilde{x}\}]_l | M_1).$$

2. *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) network M_1 and a (possibly empty) set I, with $d(l,l_i) \leq r\ \forall i \in I$, such that:*

$$M \equiv (\nu\tilde{d})(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l | \prod_{i\in I} n_i[c(\tilde{x}_i).P_i]_{l_i} \mid M_1)$$

and

$$M' \equiv (\nu\tilde{d})(n[P]_l | \prod_{i\in I} n_i[P_i\{\tilde{v}/\tilde{x}_i\}]_{l_i} \mid M_1).$$

Proof The proof follows by induction on the transition rules of Table 5. □

The structural congruence respects the transitions of Table 5.

Lemma 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 The proof is derived 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 labelled transition one.

Theorem 1 (Harmony) *Let M be a network.*

1. *If $M \to [\![M']\!]_\theta$ then there exist $N \equiv M$ and $N' \equiv M'$ such that $N \xrightarrow{\tau} [\![N']\!]_\theta$.*
2. *$M \downarrow_{c@K}$ if and only if there exist $\tilde{v}$, $R \supseteq K$ and $N \equiv M$ such that $N \xrightarrow{c!\tilde{v}@K \triangleleft R}$.*
3. *If $M \xrightarrow{\tau} [\![M']\!]_\theta$ then $M \to [\![M']\!]_\theta$.*
4. *If $M \xrightarrow{c!\tilde{v}@K \triangleleft R} [\![M']\!]_\Delta$ then $M \to [\![M']\!]_\Delta$.*

Proof See Appendix. □

3.3 Probabilistic labelled bisimilarity

We now expand the LTS rules and develop a *probabilistic labelled bisimilarity* that will be proved to be a complete characterisation of our *probabilistic behavioural congruence*. The probabilistic labelled bisimilarity is built upon the following labels:

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

Here, we write $M \xrightarrow{\alpha}_\theta N$ if $M \xrightarrow{\alpha} [\![M']\!]_\theta$ and N is in the support of $[\![M']\!]_\theta$. A *labelled execution* e of a network M is a finite (or infinite) sequence of steps:

$$M \xrightarrow{\alpha_1}_{\theta_1} M_1 \xrightarrow{\alpha_2}_{\theta_2} M_2 ... \xrightarrow{\alpha_k}_{\theta_k} M_k\,.$$

With abuse of notation, we define $Exec_M$, $last(e)$, e^j and $e\uparrow$ as for unlabeled 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 are obtained by resolving the non-determinism of both α and $[\![M]\!]_\theta$. As a consequence, a scheduler[2] in the labelled semantics is a function F associating a pair $(\alpha, [\![M]\!]_\theta) \in lbehave(last(e))$ to a finite labelled execution e. We denote by $LSched$ the set of all schedulers in the labelled semantics. Given a network M and a scheduler F, we denote by $Exec^F_M$ the set of all labelled executions starting from M and driven by F.

From a modelling point of view, we aim at distinguishing networks that differ for some observable actions, therefore ignoring internal behaviours of the nodes. Formally, this is captured by weak behavioural equivalences, that abstract over τ-actions. The notion of *weak action* is introduced below.

[2] With abuse of notation, we still use F to denote a scheduler for the labelled transition semantics.

Definition 7 (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.

We denote by $Exec^F_M(\overset{\alpha}{\Longrightarrow}, H)$ the set of executions that, starting from M and guided by F, lead to a network in the set H by performing $\overset{\alpha}{\Longrightarrow}$. Moreover, we define $Prob^F_M(\overset{\alpha}{\Longrightarrow}, H) = Prob^F_M(Exec^F_M(\overset{\alpha}{\Longrightarrow}, H))$.

Since we want our bisimilarity to be a complete characterisation of our notion of behavioural equivalence, which has been defined with respect to a restricted set of schedulers $\mathcal{F} \subseteq Sched$ on the reduction semantics, we define the set of schedulers $\hat{\mathcal{F}} \in LSched$ for the LTS corresponding to $\mathcal{F}$.

Definition 8 Given a scheduler $F \in Sched$, we denote by $\hat{F}_\mathcal{C} \subseteq LSched$ the set of schedulers $\hat{F} \in LSched$ such that $\forall M_0$, $\forall e \in Exec^{\hat{F}}_{M_0}$:

$$e = M_0 \xrightarrow{\alpha_1}_{\theta_1} M_1 ... \xrightarrow{\alpha_k}_{\theta_h} M_h$$

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

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

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

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

For a given a set $\mathcal{F} \subseteq Sched$ of schedulers, we define $\hat{\mathcal{F}}_\mathcal{C} = \bigcup_{F \in \mathcal{F}} \hat{F}_\mathcal{C}$.

Example 3 Consider the networks M_0 and N_0, and the schedulers F and F_1 of the Example 2. Let $\hat{F}_1 \in LSched$ such that

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

then, since

$$M_0 \to_\Delta M_1 \in Exec^F_{M_0}$$

the conditions of Definition 8 are satisfied by taking the empty context $C[\cdot] = [\cdot]$ and the identity function $f(i) = i$ for $i \in \{0,1\}$. Hence $\hat{F}_1 \in \hat{F}_\mathcal{C}$.

Consider now $\hat{F}_2 \in LSched$ such that

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

Since

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

with $F_1 \in F_\mathcal{C}$, by assuming the contexts $C_i[\cdot] \equiv M_i \mid \cdot$ for $i \in \{0,1\}$, and the identity function $f(i) = i$ for $i \in \{0,1\}$ we get $\hat{F}_2 \in \hat{F}_\mathcal{C}$ too. □

The following proposition holds.

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

Proof The first statement follows straightforwardly from Definition 4. To prove the second statement observe that:
$\forall F \in LSched, \forall M_0 \in \mathcal{N}$ and $\forall e = M_0 \xrightarrow{\alpha_1}_{\theta_1} M_1 ... \xrightarrow{\alpha_k}_{\theta_k} M_k \in Exec^F_{M_0}$ it is always possible to find a context $C_0[\cdot]$ and a scheduler $F' \in LSched$ such that $e' = C_0[M_0] \xrightarrow{\tau}_{\theta_1} ...C_1[M_1]... \xrightarrow{\tau}_{\theta_k} C_k[M_k] \in Exec^{F'}_{C_0[M_0]}$.
By Theorem 1, $\exists F'' \in Sched$ such that
$e'' = C_0[M_0] \rightarrow_{\theta_1} C_1[M_1]... \rightarrow_{\theta_k} C_k[M_k] \in Exec^{F''}_{C_0[M_0]}$,
meaning that $F \in \widehat{Sched}_{\mathcal{C}}$ as required. □

The notion of probabilistic labelled bisimilarity relative to a given set of schedulers is defined below. Notice that in the definition below input actions are treated differently from output and silent actions. This is due to the fact that in our model the input is not an observable action, hence two systems are considered equivalent even if they do not have the same behaviour in terms of transmission receptions.

Definition 9 (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 $F \in \hat{\mathcal{F}}_{\mathcal{C}}$ there exists a scheduler $F' \in \hat{\mathcal{F}}_{\mathcal{C}}$ such that for all α and for all classes $\mathcal{C}$ in $\mathcal{N}/\mathcal{R}$ it holds:

1. if $\alpha \neq c?\tilde{v}@l$ then $Prob^F_M(\xrightarrow{\alpha}, \mathcal{C}) = Prob^{F'}_N(\stackrel{\hat{\alpha}}{\Longrightarrow} \mathcal{C})$;
2. if $\alpha = c?\tilde{v}@l$ then either
   $Prob^F_M(\xrightarrow{\alpha}, \mathcal{C}) = Prob^{F'}_N(\stackrel{\alpha}{\Longrightarrow}, \mathcal{C})$ or
   $Prob^F_M(\xrightarrow{\alpha}, \mathcal{C}) = Prob^{F'}_N(\Longrightarrow, \mathcal{C})$.

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

3.4 A complete characterisation

Finally we prove that our probabilistic labelled bisimilarity is a complete characterisation of the probabilistic behavioural congruence of Definition 6.

Proposition 2 *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 $F \in \hat{\mathcal{F}}_{\mathcal{C}}$ there exists a scheduler $F' \in \hat{\mathcal{F}}_{\mathcal{C}}$ such that for all α and for all classes $\mathcal{C}$ in $\mathcal{N}/\mathcal{R}$ it holds:*

1. *if $\alpha \neq c?\tilde{v}@l$ then $Prob^F_M(\stackrel{\hat{\alpha}}{\Longrightarrow}, \mathcal{C}) = Prob^{F'}_N(\stackrel{\hat{\alpha}}{\Longrightarrow} \mathcal{C})$;*
2. *if $\alpha = c?\tilde{v}@l$ then either*
   $Prob^F_M(\stackrel{\hat{\alpha}}{\Longrightarrow}, \mathcal{C}) = Prob^{F'}_N(\stackrel{\alpha}{\Longrightarrow}, \mathcal{C})$ 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 now prove that our probabilistic labelled bisimilarity is a proof method for the behavioural congruence, i.e., that $\approx^{\mathcal{F}}_p$ is contained in $\cong^{\mathcal{F}}_p$.

Theorem 2 (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 show that the behavioural congruence is contained in the labelled bisimilarity.

Theorem 3 (Completeness) *Let M and N be two networks and $\mathcal{F} \subseteq Sched$. If $M \cong^{\mathcal{F}}_p N$ then $M \approx^{\mathcal{F}}_p N$.*

Proof See Appendix. □

The following result is a consequence of Theorems 2 and 3.

Theorem 4 (Characterization) *For every set $\mathcal{F} \subseteq Sched$, $\cong^{\mathcal{F}}_p = \approx^{\mathcal{F}}_p$.*

4 Energy Consumption Estimation

In this section, based on the labelled transition semantics, we define a preorder over networks to contrast the average energy cost of different networks but exhibiting the same connectivity behaviour relative to a specific set of schedulers $\mathcal{F}$. Formally, an energy cost is associated with labelled transitions as follows:

$$\mathbf{Cost}(M, N) = \begin{cases} r & \text{if } M \xrightarrow{c_L![l,r]} [\![N]\!]_\Delta \text{ for some } c, L, \tilde{v}, l \\ 0 & \text{otherwise.} \end{cases}$$

This can be read as: the energy cost to reach N from M in one single step is r if M can reach N after firing on a channel of radius[3] r independently from the fact that the transmitted message is observable or not (or even lost). Moreover, for a given execution $e = M_0 \xrightarrow{\alpha_1}_{\theta_1} M_1 ... \xrightarrow{\alpha_k}_{\theta_k} M_k$ we define

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

[3] Note that considering the radius of the communication channel as the energy cost of the transmitted data is standard (see, e.g., [40,41]).

Given a set of networks H, we denote by $Paths_M^F(H)$ the set of all executions from M ending in H and driven by F which are not a prefix of any other execution ending in H. More formally,

$$Paths_M^F(H) = \{e \in Exec_M^F(H) \mid last(e) \in H \text{ and } \forall e' \text{ such that } e \text{ is a prefix of } e',\ e' \notin Paths_M^F(H)\}.$$

We are now in position to define the average energy cost of reaching a set of networks H from the initial network M according to a scheduler F.

Definition 10 Let H be a set of networks. The average energy cost of reaching H from M according to the scheduler F is

$$\mathbf{Cost}_M^F(H) = \frac{\sum_{e \in Paths_M^F(H)} \mathbf{Cost}(e) \times P_M^F(e)}{\sum_{e \in Paths_M^F(H)} P_M^F(e)}.$$

Basically, the average cost is computed by weighting the cost of each execution by its probability according to F and normalized by the overall probability of reaching H. The following definition provides an efficient method to perform both qualitative and quantitative analyses of mobile networks.

Definition 11 Let $\mathcal{H}$ be a countable set of sets of networks and let $\mathcal{F} \subseteq Sched$ a set of schedulers. We say that N is *more energy efficient than* M relative to $\mathcal{H}$ and $\mathcal{F}$, denoted

$$N \sqsubseteq_{\langle \mathcal{H}, \mathcal{F} \rangle} M,$$

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

5 Performance evaluation of a location based routing protocol

In this section we consider a network of nodes with mobility and using the Location Aided Routing protocol (LAR) [7]. LAR aims at reducing the number of the packet floods with respect to what is observable in other protocols such as the AODV [36]. This is achieved by assuming that the nodes are aware of their own absolute or relative positions, e.g., because they are equipped with a GPS device [42] or because they are able to derive their distances from a set of fixed nodes. With respect to the analysis of LAR presented in [13], here we consider a quantitative approach that allows us to study the energy efficiency of LAR with respect to AODV under different scenarios. In order to carry out this comparison, we encode the AODV and LAR models described by means of the process calculus that we have defined into a PRISM program [16] and we perform a statistical model checking to estimate the energy consumptions of the protocols. We also prove that AODV and LAR are behaviourally equivalent, i.e., under the modelling assumptions, a packet is correctly delivered by AODV if and only if it is correctly delivered by LAR.

5.1 Protocol Description

In very large mobile networks using flooding strategies such as AODV [36] may be very expensive in terms of number of sent packets and hence of node energy consumption. The LAR protocol requires the sender node to guess the location of the destination and therefore it can avoid the use of flooding strategies. The destination node's location is inferred by its location that has been transmitted during the latest packet exchange and an assumption on the maximum node speed.

5.2 Simple flooding: description

We briefly recall some basic notion on flooding based algorithms. In these algorithms the route discovery is carried out by exchanging three types of packets [43]:

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

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

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

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

  where S, Bid and D have the same meaning of before, $\texttt{seq\#}_D$ is the sequence number maintained at the destination, `hop_counter` is the number of hops to reach the destination and `Lifetime` is the time to live associated with the route.
- *Route Error* packet (RERR) has the form:

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

  where S, D and $\texttt{seq\#}_D$ are defined as before and are used to handle errors in the protocol route discovery.

In the flooding algorithm, a node that wants to discover a route to a destination first broadcasts a RREQ packet. When the destination receives the RREQ it replies with a RREP which is forwarded to the source in unicast mode toward the same path used by the RREQ. A timeout mechanism is adopted to avoid nodes starvation.

5.3 The LAR algorithm

The LAR protocol differs from the basic flooding because it does not perform a complete network broadcast in the route discovery phase, but it limits the area in which the RREQ packets are transmitted to where the destination node is expected to be found (*Expected Zone*). If this strategy does not work, then a complete flood is performed.

The expected zone is determined as follow: suppose the source node S knows the location l_1 of the destination node D at time t, and D moves with a speed v. S expects to find D is circle area with center l_1 and radius $v(t'-t)$, where t' is the epoch in which the transmission is being done. In the cases in which S does not have any information about the locations of D the *Expected Zone* coincides with the entire network.

Packets are forwarded only by the nodes that lies in the *Request Zone* which is defined by the sender. There is a trade off in the specification of the *Request Zone*: smaller ones reduce the number of packets required to discover the route to the destination, whereas large ones reduce the latency of the route discovery phase. In the literature, several strategies have been proposed to define the *Request Zone*, we present that called *LAR Scheme 1*. This defines the *Request Zone* as the smallest rectangle containing both the *Expected Zone* and the position of the source node (see Fig. 1).

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

$$A :\rightarrow (X_S, Y_D + R) \quad B :\rightarrow (X_D + R, Y_D + R)$$
$$C :\rightarrow (X_D + R, Y_S) \quad D :\rightarrow (X_S, Y_S)$$

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

$$A :\rightarrow (X_D - R, Y_D + R) \quad B :\rightarrow (X_D + R, Y_D + R)$$
$$C :\rightarrow (X_D - R, Y_D - R) \quad D :\rightarrow (X_D + R, Y_D - R)$$

5.4 Modelling the network

We encode the simple flooding and the LAR protocols using our calculus. We consider a 80×100 metres area of 35 mobile nodes. We omit the implementation details about how the *Expected Zone* and *Request Zone* are determined according to the specifications of LAR Scheme 1.

We use the following 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;
- `find_path`(n) = `true` if there exists a valid path for n in the route table of the node executing the process;
- `newBid`: generates a new unique Bid identifier for a packet;
- `lastBid`: returns the latest generated Bid identifier;
- `control`(Bid) = `true` if the request associated with Bid has been already received by the node executing the process.

Each record in the nodes' *routing tables* is structured as follows:

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

with:

- d: the destination name
- $\texttt{seq\#}_d$: the sequence number associated with the route to d
- $\texttt{next\_hop}_d$: name of the next node to reach d
- $\texttt{hopcount}_d$: number of hops to reach d
- $\texttt{loc}_d$: the last location known of d
- $\texttt{v}_d$: expected speed of d
- `timeout`: time to leave of the record

The *request table* is used by the nodes to store the history of all the requests that have been previously processed by the nodes. This prevents the creation of loops during the route request forwarding phase. For the sake of simplicity, we assume that all the nodes share a common transmission radius $r = 15$ metres.

We define the following model: $N = (\nu c)(n[P]_l \mid \prod_{i \in I} n_i[Q\_SIMPLE]_{l_i})$ that represents a node n which moves among the locations in $\{16, 23, 30\}$ and broadcasts a route request according to the flooding protocol aimed to find a path to n_7. Fig. 2 (a) shows the location of the nodes in the network $\prod_{i \in I} n_i$. Moreover, consider $M = (\nu c)(n[P]_l \mid \prod_{i \in I} n_i[Q\_LAR1]_{l_i})$ which models the same network in which the nodes in I implements the LAR protocol with (Scheme 1). The DTMC that describes the movements of node n_i is identified by the matrix J^{n_i}:

	l_{n_i}	k_{n_i}
l_{n_i}	0.2	0.8
k_{n_i}	0.8	0.2

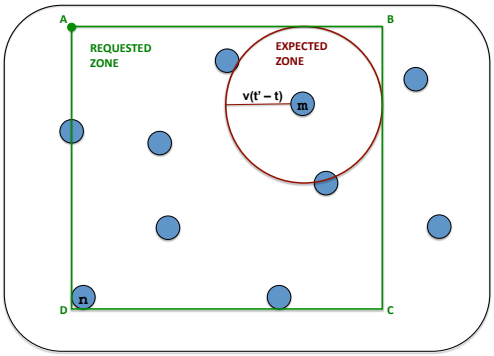
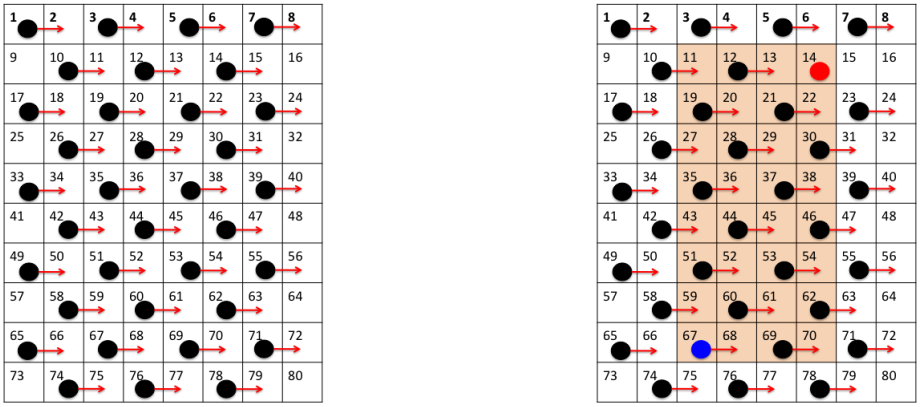


Fig. 1: *Expected* and *Request Zones* in the LAR protocol



(a) Flooding Area

(b) Location-Aided Routing Area

Fig. 2: Topology of the network

where l_{n_i} and k_{n_i} are adjacent locations (Fig. 2 (b)).

Node n behaves as follows: it broadcasts a RREQ packet with destination node n_7 and waits for a RREP packet. We use the operator $\oplus$ to model the timeout. Notice that, in our calculus, the non-deterministic choice and can be implemented with the parallel composition and the restriction operator in the standard way. When the timeout expires n broadcasts a new RREQ with the same destination. Let

$$P = \bar{c}_{\mathbf{Loc},r}\langle(\mathtt{rreq}, n, \mathtt{newBid}, n_7, \mathtt{Request\_Zone}, \mathtt{seq}\#_n, 0)\rangle.P'$$

and

$$P' = P \oplus c(x_1,x_2,x_3,x_4,x_5,x_6,x_7).\,[x_1 = \mathtt{rrep}][x_2 = n][x_3 = \mathtt{lastBid}][x_4 = m]\,[\mathtt{geq}(\mathtt{hop\_count}_{n_7}, x_7)]\bar{\mathtt{ok}}_{\mathbf{gps},r}\langle\mathtt{route\_found}\rangle.P'$$

where x_7 = `hop_count` in the RREP packet received. For modelling purpose, in order to be able to observe a correct route discovery by n, we assume that when this event occurs n transmits on the fictitious channel `ok` an acknowledgement packet. With this simplification, we say that two networks are probabilistically equivalent with respect to their ability on finding a route to n_7 if we observe this transmission with the same probability.

Hereafter, we use $X \in \{SIMPLE, LAR1\}$ to denote the simple flooding or LAR Scheme 1.
The $RREQ\_SIMPLE$ and the $RREQ\_LAR1$ subprocess are defined as shown by Table 6.

In order to compare the behaviour of the protocols, we restrict the set of admissible schedulers $\mathcal{F} \subseteq Sched$ to those that satisfy the following conditions:

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 after every transmission.

Condition 1 on $\mathcal{F}$ derives from the protocol specification and indeed it is usually set according to the max-

$$
\begin{array}{l}
Q\_X = c(x_1,x_2,x_3,x_4,x_5,x_6,x_7).[x_1 = \texttt{rreq}]([\texttt{control}(x_3) = \texttt{false}] \\
\quad ([x_4 = \texttt{self}]\bar{c}_{\texttt{next\_hop}_{x_2},r}\langle(\texttt{rrep}, s, Bid, d, \texttt{seq\#}_s, \texttt{hop\_counter})\rangle. \\
\quad Q\_X, RREQ\_X\langle\tilde{x}\rangle), Q\_X), [x_1 = \texttt{rrep}]([\texttt{x}_2 = \texttt{self}] \\
\quad \bar{ud}_{\texttt{gps},r}\langle x_2,x_3,x_4,x_5,x_6,x_7\rangle, \\
\quad \bar{c}_{\texttt{next\_hop}_{x_2},r}\langle(\texttt{rrep}, s, Bid, d, \texttt{seq\#}_s, \texttt{hop\_counter})\rangle.Q\_X), \\
\quad [x_1 = \texttt{rerr}]\texttt{unable}(x_4).Q\_X, Q\_X \\
\\
RREQ\_SIMPLE\langle(\texttt{rreq}, s, Bid, d, \texttt{seq\#}_s, \texttt{hop\_counter})\rangle = \\
\quad [\texttt{find\_path}(d) = \texttt{true}]. \\
\quad \bar{c}_{\texttt{next\_hop}_d,r}\langle(\texttt{rrep}, s, Bid, d, \texttt{seq\#}_d, \texttt{hop\_counter} + 1 + \texttt{hopcount}_d, \texttt{timeout})\rangle, \\
\quad \bar{c}_{\mathbf{Loc},r}\langle(\texttt{rreq}, s, Bid, d, \texttt{seq\#}_s, (\texttt{hop\_counter}) + 1)\rangle.Q\_SIMPLE \\
\\
RREQ\_LAR1\langle(\texttt{rreq}, s, Bid, d, \texttt{Request\_Zone}, \texttt{seq\#}_s, \texttt{hop\_counter})\rangle = \\
\quad ([\texttt{inside}(\texttt{gps}, \texttt{Request\_Zone}) = true][\texttt{find\_path}(d) = \texttt{true}] \\
\quad \bar{c}_{\texttt{next\_hop}_d,r}\langle(\texttt{rrep}, s, Bid, d, \texttt{seq\#}_d, \texttt{hop\_counter} + 1 + \texttt{hopcount}_d, \texttt{timeout})\rangle, \\
\quad \bar{c}_{\texttt{Request\_Zone},r}\langle(\texttt{rreq}, s, Bid, d, \texttt{Request\_Zone}, \texttt{seq\#}_s, (\texttt{hop\_counter}) + 1)\rangle)).Q\_LAR1
\end{array}
$$

Table 6: Process specifications used in the case study of Section 5

imum delay that a packet spends to cover the longest distance in the network. Informally, this condition excludes the schedulers associate to timeout which are set so short that a packet cannot be received in time or those that waits for the reply indefinitely long. Condition 2 has been introduced to ensure that mobility is taken into account in the comparison.

Proposition 3 states the functional equivalence between the AODV and LAR protocols. It holds for all networks M and N implementing the LAR and AODV protocols as described above with arbitrary number of nodes, locations and node distances provided that the DTMC modelling the mobility is ergodic on the set of locations.

Proposition 3 (Functional equivalence of LAR and AODV) *Let M and N be two networks implementing the LAR and AODV protocols, respectively. Let $\mathcal{M} = \{\bar{M} : M \rightarrow^* \bar{M}\} \cup \{\bar{N} : N \rightarrow^* \bar{N}\}$ and $\mathcal{F}$ be the set of admissible schedulers defined as above. A sufficient condition for $N \approx_p^{\mathcal{F}} M$ is that the Markov chains $\mathbf{J}^{n_i}$ associated with the mobile nodes n_i $(i \in I)$ are ergodic.*

Proof We have to find a relation containing the pair (M, N) that is a probabilistic bisimulation relative to $\mathcal{F}$. Let us consider $Z_i \in \{RREQ, Q\}$, $\bar{P} \in \{P' : P \rightarrow^* P'\}$ and the relation

$$
\begin{array}{l}
\mathcal{R} = \{(n[\bar{P}]_l \mid \prod_{i\in I} n_i[Z_i\_SIMPLE]_{l_i}, n[\bar{P}]_l \mid \\
\qquad \prod_{i\in I} n_i[Z_i\_LAR1]_{l_i}) : \\
\qquad N \rightarrow^* n[\bar{P}]_l \mid \prod_{i\in I} n_i[Z_i\_SIMPLE]_{l_i}\}.
\end{array}
$$

In order to prove that $\mathcal{R} \subseteq \approx_p^{\mathcal{F}}$ we have to show that, for all pairs $(\bar{N}, \bar{M}) \in \mathcal{R}$ and for all schedulers $F \in \hat{\mathcal{F}}_{\mathcal{C}}$ there exists a scheduler $F' \in \hat{\mathcal{F}}_{\mathcal{C}}$ such that for all α and for all classes $\mathcal{C}$ in $\mathcal{N}/\mathcal{R}$ it holds:

1. if $\alpha \neq c?\tilde{v}@l$ then
   $Prob_{\bar{N}}^{F}(\xrightarrow{\alpha}, \mathcal{C}) = Prob_{\bar{M}}^{F'}(\stackrel{\hat{\alpha}}{\Longrightarrow} \mathcal{C})$;
2. if $\alpha = c?\tilde{v}@l$ then either
   $Prob_{\bar{N}}^{F}(\xrightarrow{\alpha}, \mathcal{C}) = Prob_{\bar{M}}^{F'}(\stackrel{\alpha}{\Longrightarrow}, \mathcal{C})$ or
   $Prob_{\bar{N}}^{F}(\xrightarrow{\alpha}, \mathcal{C}) = Prob_{\bar{M}}^{F'}(\Longrightarrow, \mathcal{C})$.

We start from τ actions and consider $\bar{N} \xrightarrow{\tau} [\![\bar{N}']\!]_\theta$. Then, $\forall \mathcal{C} \in \mathcal{N}/\mathcal{R}$, we have:

$$Prob_{\bar{N}}(\xrightarrow{\tau}, \mathcal{C}) = \sum\nolimits_{\hat{N} \in spt([\![\bar{N}']\!]_\theta) \cap \mathcal{C}} [\![\bar{N}']\!]_\theta(\hat{N})\,.$$

If the action is due to the application of rule (Move) we are done, because, for each pair $(\bar{N}, \bar{M}) \in \mathcal{R}$, $\bar{M}$ can perform exactly the same movements as $\bar{N}$, hence there will exist $F' \in \hat{\mathcal{F}}_{\mathcal{C}}$ such that: $Prob_{\bar{N}}^{F}(\xrightarrow{\tau}, \mathcal{C}) = Prob_{\bar{M}}^{F'}(\xrightarrow{\tau} \mathcal{C})$, and we are done.

If the action is the result of the application of rule (Lose), by applying rule (Bcast) backwardly we get $\bar{N} \xrightarrow{c_K!\tilde{v}[l,r]} [\![\bar{N}']\!]_\Delta$.

If $l \in$ `Request_Zone` then we are done, because, by the analysis of the process P_LAR1 with respect to P_SIMPLE we note that the protocol packets are forwarded exactly in the same way inside the `RequestZone`.

If $l \notin$ `Request_Zone`, then $\bar{M} \not\xrightarrow{c_K!\tilde{v}[l,r]}$ because the routing protocol packets are forwarded only inside the `Request_Zone`. However, this does not mean that $\bar{M}$ will not reach an equivalent state with the same probability. By the initial hypothesis that all the Markov

distance	flooding	LAR
56,56	11,77	5,94
72,11	15,55	8,16
89,44	15,14	14,87

Fig. 4: Estimates of the expected energy cost w.r.t. sent packets per successfully transmission.

matrices are ergodic, $\bar{M}$ can enter the `Request_Zone` with probability 1, send the message, and come back to the previous location again with probability 1, and we get $Prob^F_{\bar{N}}(\xrightarrow{\tau}, \mathcal{C}) = 1 = Prob^F_{\bar{M}}(\Longrightarrow, \mathcal{C})$ as required.

As concerns the input and the observable actions the proof is trivial, since the input actions are the same for both protocols, and we applied the restriction to channel c, hence the only observable output is the transmission of `route_found` through the channel `ok` by the node n, which behaves in the same way for both protocols. □

Given that the two networks M and N defined at the beginning of this section are functionally equivalent, we compare their energy efficiency by simulation. In order to carry out the simulations we resort to the statistical model checker implemented in PRISM [16]. This technique is commonly used when dealing with models with large state spaces. The simulation model for the PRISM has been automatically generated by the tool introduced in [44].

We have compared the two different networks with the sender node n located in each of the locations in the set $\{16, 23, 30\}$.

The simulations have been performed with an average of 10000 independent experiments, a maximum confidence interval width of 1% of the estimated measure based on 95% of confidence.

The plot (see Fig. 3) shows the relation among the distance between sender and receiver and the energy consumption of AODV and LAR expressed in terms of number of sent packets for each successful transmission. For larger distances, since a larger *Request Zone* is involved, using LAR protocol still requires a large set of nodes to forward the message, while for smaller distances the improvement brought by the protocol is more evident, since the *Request Zone* is smaller, drastically reducing the number of retransmissions. This supports the intuitive idea that LAR protocol is useful especially in the cases where the expected distance between the sender and the receiver is small. In Fig. 4 we show the numerical comparison between the LAR protocol and the AODV for the considered scenarios.

6 Analysing the SW-ARQ and GBN-ARQ Protocols

In the following we briefly recall the salient features of SW-ARQ and GBN-ARQ protocols. In SW-ARQ protocol, the sender pushes a packet into the channel with a delay that is given by ratio between the packet size and the channel bandwidth (pushing time). Once the packet is in the channel we observe two delays: one is that required to reach the destination and the other one is that required for the acknowledge packet (ACK) to go back to the transmitter. The sum of the two is known as the round trip time. In SW-ARQ protocol the sender sends a packet only once the acknowledge of the previous one has been received. If the round trip time (or an upper bound) is known by the protocol designer, a possible error in the transmission is detected by a timeout mechanism, i.e., if the sender does not receive an ACK from the receiver before a deadline, then it assumes that an error occurred and sends again the same packet. If the round trip time is much higher than the pushing time, then SW-ARQ protocols are very inefficient and exploit only a minimal part of the channel capacity. With respect to SW protocols, GBN takes advantage of the pipelining of the packets, i.e., a sequence of n packets can be sent without receiving any confirmation. This widely used technique is known to highly improve the throughput of the sender, but it is expensive from the energy consumption point of view (see, e.g., [45]) since correctly received packets may be required to be resent. Indeed, once the sender realizes that a packet p has not been received (using a timeout), it has to resend all the packets already sent starting from p. In this way, it can be shown that throughput is really improved and the protocol can use the full channel capacity.

6.1 Assumptions on the models

In this case study, we consider a single transmitter node using ARQ-based error recovery protocol to communicate with a receiver node over a wireless channel. Transmissions occur in fixed-size time slots whose size is the time required by the sender to push a packet into the channel. We assume the round trip time to be a multiple of the time slot. For both SW and GBN protocols, the transmitter continuously sends packets until it detects a transmission error. Notice that although in actual implementations of the ARQ protocols errors are usually detected by means of a timeout mechanism, in this context we use negative-acknowledge (NACK) feedbacks which simplify the protocol encoding and are equivalent for the analysis purposes if we assume to

Fig. 3: Plot of the expected energy cost w.r.t. sent packets per succesful transmission.

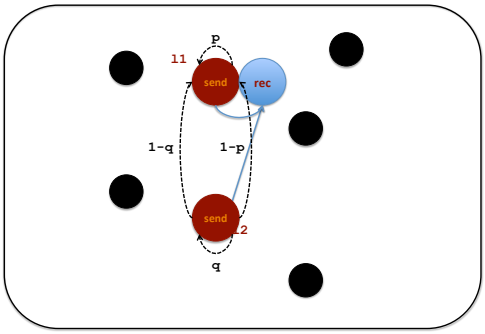


Fig. 5: Topology of the network and mobility of the sender

know the number of slots that the round trip time consists of. Here, we consider an error-free feedback channel [4] and assume that the ACK or NACK of each transmitted packet arrives at the sender node one slot after the beginning of its transmission slot. Therefore, the feedback of a packet is received exactly after its transmission for the SW-protocol and in case of a failure (NACK), the packet is automatically resent. Instead for the GBN protocol, a feedback for the ith packet arrives exactly after the transmission of the $(i+n-1)$th packet and in case of a failure the transmission restarts from the ith packet. We model both SW-ARQ and GBN-ARQ-based protocols for a communication channel of capacity $n = 3$ in our framework. Observe that in this way we do not take into account the round trip time for SW-ARQ protocols, however this does not affect the analysis that we will carry out later, i.e., the expected energy cost for each packed correctly received. We consider a unique static receiver $rec < 0, I >$ where I denotes the identity matrix. We model the transmitter as a mobile node $send$ $(< r, J^s >)$ whose reachable locations are l_1, which represents the "good state" of the channel, where the receiver lies within the transmission radius of the channel and l_2 the "bad state", where the destination is no longer reachable (see Fig. 5). The mobility of the sender is modelled by the two state Markov chain with the following transition probability matrix

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

where p and q are the probabilities of the stability of the node in two successive time slots in its good and bad states, respectively.

6.2 Modelling the Protocols

In our analysis, we assume that the energy consumption of the feedback messages is negligible. Therefore, they are sent over channels with zero radius. For this reason the static receiver rec is located at l_1, i.e., at the same location of the sender in its good state, so that the feedback will be received with no cost. Note that the sender still transmits over channels with radius r

[4] A very standard assumption [45].

and thus it consumes an amount of energy equal to r for each fired packet.

The process executed by rec, the receiver node, is the same for both protocols and modelled as the process

$$REC\langle i\rangle = c^{(i)}(x).\bar{c}_{l_1,0}\langle ACK(i)\rangle.REC\langle i+1\rangle$$

which, upon receiving packet p_i over the channel $c^{(i)}$, sends $ACK(i)$ over the channel c and waits for the next packet on $c^{(i+1)}$.

For each channel $c^{(i)}$, we use a static auxiliary node $b_i(\langle 0, I\rangle)$ located at l_2, the bad state of the sender, capturing bad transmissions over $c^{(i)}$. It executes the following process which upon receiving packet p_i over the channel $c^{(i)}$, sends $NACK(i)$ over the channel c:

$$BAD\langle i\rangle = c^{(i)}(x).\bar{c}_{l_2,0}\langle NACK(i)\rangle.BAD\langle i\rangle.$$

6.2.1 GBN-ARQ.

Now we introduce the full model of the protocol GBN-ARQ. We start by modelling its sender node. Recall that, as a simplifying assumption, the channel capacity is 3. It executes the following process:

$$GB\langle i\rangle = \bar{c}^{(i)}_{l_1,r}\langle p_i\rangle.c(x_1).\bar{c}^{(i+1)}_{l_1,r}\langle p_{i+1}\rangle.c(x_2).\bar{c}^{(i+2)}_{l_1,r}\langle p_{i+2}\rangle.\\ c(x_3)[x_1 = NACK(i)]GB\langle i\rangle, SEND\langle i+3, x_2, x_3\rangle$$

where the process $SEND$ is defined as follows.

$$SEND\langle i, x, y\rangle = \bar{c}^{(i)}_{l_1,r}\langle p_i\rangle.c(z).[x = NACK(i-2)]\\ GB\langle i-2\rangle, SEND\langle i+1, y, z\rangle.$$

Though that the feedback of a packet is received after the transmission of its two successors, for practical reason, we read a feedback of a packet right after sending it. Indeed, since we do not want feedback to be costly, both sender and receiver must be located at the same place when the feedback is sent. However, the sender node will verify it only after having sent the following two packets.

Recall that the receiver node in our modelling above, reads each packet p_i on its specific channel $c^{(i)}$. Thus, in the GBN, if the transmitter sends p_1 while being in its good state, then moves to bad and sends p_2 and finally moves back to the good state and sends p_3, then the later packet will not be read by the receiver as it is blocked on $c^{(2)}$. Then, the firing on $c^{(3)}$ is lost and this models the fact that packets sent after a bad packet is just a wasting of energy. But since the sender process $GB\langle i\rangle$ is blocked on the feedback channel c, we introduce a static auxiliary node $lose(\langle 0, I\rangle)$ located at l_1 and executing the process:

$$WAST = \bar{c}_{\emptyset,0}\langle LOST\rangle.WAST$$

6.2.2 SW-ARQ.

Now on to the SW-ARQ-based protocol. This is very simple since it always sends one packet and waits for its feedback. The sender process is defined as follows.

$$SW\langle i\rangle = \bar{c}^{(i)}_{l_1,r}\langle p_i\rangle.\\ c(x).[x = NACK(i)]SW\langle i\rangle, SW\langle i+1\rangle.$$

The full protocols are then modelled as:

$$GBN = (\nu c^{(1)}, c^{(2)}...)(send[GB\langle 1\rangle]_{l_1}\\ \mid rec[REC\langle 1\rangle]_{l_1} \mid lose[WAST]_{l_1} \mid\\ \textstyle\prod_{i\geq 1} b_i[BAD\langle i\rangle]_{l_2})$$

$$SW = (\nu c^{(1)}, c^{(2)}...)(send[SW\langle 1\rangle]_{l_1} \mid rec[REC\langle 1\rangle]_{l_1}\\ \mid \textstyle\prod_{i\in I} b_i[BAD\langle i\rangle]_{l_2}).$$

6.3 Measuring the Energy Cost of the Protocols.

This section analyzes the energy consumption of the above ARQ-based protocols. In order to compare the observational behaviours of the protocols, we assume that the communications over the feedback channel are observable for any observer node located at l_1. Thus the protocols are equivalent with respect to a set of schedulers $\mathcal{F}$ if for all schedulers F in $\mathcal{F}$ driving one of the protocols, there exists a scheduler F' in $\mathcal{F}$ driving the other one such that both protocols correctly transmit the same packets with the same probabilities. We consider the following set of schedulers denoted $\mathcal{F}_{alt}$ which:

1. always alternates between sending packets and node's movement so that at each interaction of the transmitter with the channel, the later can be either good or bad;
2. gives priority to acknowledgment actions (ACK and NACK) to model the standard assumption of an error-free feedback channel;
3. allows interaction with the outside environment only through its observable actions so that we capture exactly the observable behaviour of the protocol.

Notice that the assumptions on the schedulers would be stricter if one desires to carry out an analysis of the throughput. If we consider the set of schedulers $\mathcal{F}_{alt}$, we can prove that the SW-ARQ protocol is more energy efficient of the GBN-ARQ one. This follows from the following results.

Proposition 4 $GBN \approx_p^{\mathcal{F}_{alt}} SW$.

Proof We give here a sketch. For each sender's window size we will choose, the only observable actions are the acknowledgments sent by the static node rec. All other actions are silent, since we apply the restriction on each

$c^{(i)}$. For all $i \geq 1$ $rec[REC\langle i\rangle]_{l_1}$ sends the acknowledgment $ACK(i)$ if and only if the relative packet p_i has been correctly received, hence, all the executions performed by GBN and SW are of the form:

$$\Longrightarrow \xrightarrow{c!ACK(1)@\{l1\}\triangleleft\{l1\}} \Longrightarrow \xrightarrow{c!ACK(2)@\{l1\}\triangleleft\{l1\}} \Longrightarrow \ldots$$

Since the number of transmissions performed by the sender do not affect the probabilities, the bisimulation between the two protocols can be proved. □

We compare the energy efficiency with respect to the set $\mathcal{H} = \{H_k \mid k \geq 1\}$ where H_k means that all the packets up to k have been correctly transmitted and is defined as $H_k = H_k^1 \cup H_K^2$ where

$$\begin{aligned} H_k^1 = \{M \mid M \equiv\ & send[\bar{c}_{l_1,r}^{(k+1)}\langle p_{k+1}\rangle.P]_{l_1} \mid \\ & rec[REC\langle k+1\rangle]_{l_1} \mid lose[WAST]_{l_1} \mid \\ & \textstyle\prod_{i\geq 1} b_i[BAD\langle i\rangle]_{l_2}\} \end{aligned}$$

for some process P and

$$\begin{aligned} H_k^2 = \{N \mid N \equiv\ & send[SW\langle i+1\rangle]_{l_1} \mid \\ & rec[REC\langle k+1\rangle]_{l_1} \mid \textstyle\prod_{i\in I} b_i[BAD\langle i\rangle]_{l_2}\}. \end{aligned}$$

Then, we compute the energy consumption of the protocols assuming that we start by a move action at the good state so that the first message could be lost if it moves to the bad state[5]. The results are summarized in the following propositions and illustrated in Fig. 6.

Proposition 5 *If $q \neq 1$ then for all $F \in \mathcal{F}_{alt}$*

$$\mathbf{Cost}_{SW}^{F}(H_k) = \left(1 + \frac{1-p}{1-q}\right) kr.$$

Proposition 6 *If $q \neq 1$ then for all $F \in \mathcal{F}_{alt}$*

$$\mathbf{Cost}_{GBN}^{F}(H_k) = kr\Big(p + \tfrac{(p-1)}{(-1+q)(1+p^2-q+q^2-p+2pq)} \cdot \tfrac{1-2p^2+2p^2q+4q-4q^2+2q^3+2p-6pq+4pq^2}{-p^2+p^2+(-p+pq)(-1+2q)+q(2+-2q+q^2)}\Big).$$

These results can be derived by applying the Chapman-Kolmogorov's forward equations to compute the probability of consecutive failures in the sending of the same packet. Each failure (except the first) causes the waste of a number of sent packets equals to the window size. Note that the number of wasted windows has a geometric distribution. Then, the mean of total packets sent to obtain a success, can be straightforwardly derived.

To conclude this section, we note that while both protocols increasingly enjoy bad performance in terms of energy consumption when the channel deteriorates, i.e., when q is increasing (see Fig. 6-(a) and 6-(b)), the GBN protocol deteriorates faster. Indeed, as illustrated by Fig. 6-(c) as the channel deteriorates the additional energy required by GBN protocol to correctly transmit the same number of packets increases to infinite. Thus, the gain of having a high throughput results in a very high energy consumption.

The next theorem follows by Propositions 4, 5 and 6.

Theorem 5 *It holds that $SW \sqsubseteq_{\langle\mathcal{H},\mathcal{F}_{alt}\rangle} GBN$.*

[5] The analysis for the other case is similar.

7 Conclusion

Ad-hoc network is a new area of mobile communication networks that has attracted significant attention due to its challenging problems. The main goal of our work is to provide a formal model to reason about the problem of limiting the power consumption of communications while maintaining acceptable performances. Indeed, one of the most critical challenges in managing mobile ad-hoc networks is actually to find a good trade-off between network connectivity and power saving.

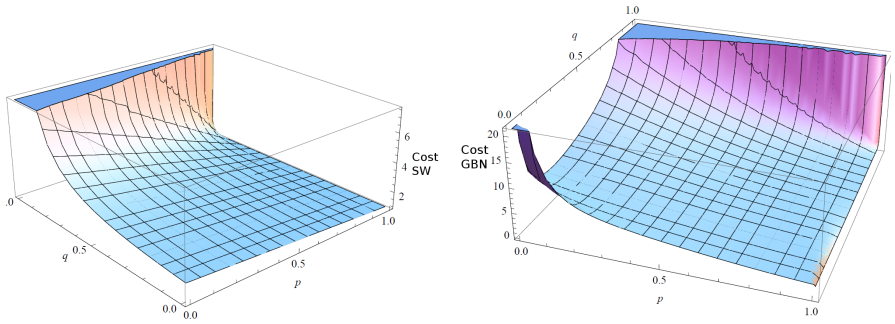
Even though not all the devices have the ability of adjusting their transmission power, modern technologies are quickly evolving, and there exist devices that are enabled to choose among two or more different power levels. For this reason many researchers have proposed algorithms and protocols with the aim of providing a way to decide the best transmission power for node communications in a given network [46,47], or to develop energy-aware routing protocols [48,49].

In this paper, we presented the Probabilistic EBUM calculus which, due to its characteristics of modelling broadcast, multicast and unicast communications and also modelling the ability of a node to change its transmission power, results to be a valid formal model for the analysis, evaluation and comparison of energy-aware protocols and algorithms specifically developed for wireless ad-hoc networks. The model we presented can clearly be extended with different metrics for measuring, e.g., the level of interference or the number of collisions and losses. Moreover, it provides a basis for the definition of other verification techniques, like e.g., bisimulation-based preorders (see [50]) which integrate both observational properties and quantitative ones.

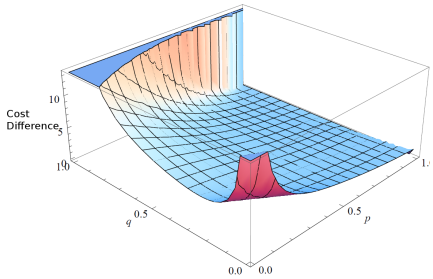
We have shown that our calculus can be implemented within the model checker PRISM. Then both exact analysis and discrete-event simulation become available for the performance evaluation of the models defined in terms of the Probabilistic EBUM calculus.

Acknowledgments

Work partially supported by the Italian MIUR - PRIN Project *CINA: Compositionality, Interaction, Negotiation and Autonomicity.*



(a) SW protocol (b) GBN protocol



(c) $\text{cost}_{GBN}(p,q) - \text{cost}_{SW}(p,q)$

Fig. 6: Energy cost functions for SW and GBN and their comparison.

References

1. Singh S, Woo M, Raghavendra C. Power-aware Routing in Mobile ad Hoc Networks. *Proc. of the 4th annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'98)*, ACM, 1998; 181–190.
2. Zorzi M, Rao R. Error Control and Energy Consumption in Communications for Nomadic Computing. *IEEE Transactions on Computers* 1997; **46**(3):279–289.
3. Sarkar S, Majumder K. A survey on power aware routing protocols for mobile ad-hoc network. *Proceedings of the International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2013*, *Advances in Intelligent Systems and Computing*, vol. 247. Springer, 2014; 313–320.
4. Papadopoulos A, Navarra A, McCann JA, Pinotti CM. Vibe: An energy efficient routing protocol for dense and mobile sensor networks. *Journal of Network and Computer Applications* 2012; **35**(4):1177 – 1190.
5. Han SW, Jeong IS, Kang SH. Low latency and energy efficient routing tree for wireless sensor networks with multiple mobile sinks. *Journal of Network and Computer Applications* 2013; **36**(1):156 – 166.
6. Abreu C, Ricardo M, Mendes P. Energy-aware routing for biomedical wireless sensor networks. *Journal of Network and Computer Applications* 2014; **40**:270 – 278.
7. Ko Y, Vaidya N. Locationaided Routing (LAR) in Mobile Ad hoc Networks. *Wireless Networks* 2000; **6**:307–321.
8. Stojmenovic I, Lin X. Power-aware Localized Routing in Wireless Networks. *IEEE Transactions on Parallel and Distributed Systems* 2001; **12**(11):1122–1133.
9. Gallina L, Hamadou S, Marin A, Rossi S. A Probabilistic Energy-aware Model for Mobile Ad-hoc Networks. *Proc. of the 18th International Conference on Analytical and Stochastic Modelling Techniques and Applications (ASMTA'11)*, *LNCS*, vol. 6751, Springer-Verlag, 2011; 316–330.
10. Gallina L, Rossi S. A Process Calculus for Energy-aware Multicast Communications of Mobile ad-hoc Networks. *Wireless Communications and Mobile Computing* 2013; **13**(3):296–312.
11. Lanese I, Sangiorgi D. An Operational Semantics for a Calculus for Wireless Systems. *Theoretical Computer Science* 2010; **411**(19):1928–1948.
12. Merro M. An Observational Theory for Mobile Ad Hoc Networks. *Information and Computation* 2009; **207**(2):194–208.
13. Bugliesi M, Gallina L, Hamadou S, A M, Rossi S. Behavioral equivalences and interference metrics for mobile ad-hoc networks. *Performance Evaluation* 2014; **73**:41–72.
14. Gallina L, Rossi S. A Calculus for Power-aware Multicast Communications in Ad-hoc Networks. *Proc. of the 6th IFIP International Conference on Theoretical Computer Science (TCS'10)*, Springer, 2010; 20–31.
15. Gallina L, Rossi S. Sender- and Receiver-centered Interference in Wireless ad-hoc Networks. *Proc. of IFIP Wireless Days 2010*, IEEE, 2010.
16. Kwiatkowska MZ, Norman G, Parker D. PRISM 4.0: Verification of Probabilistic Real-time Systems. *CAV*, 2011; 585–591.
17. Milner R, Sangiorgi D. Barbed Bisimulation. *Proc. of International Colloquium on Automata, Languages and Programming (ICALP '92)*, *LNCS*, vol. 623, Springer-Verlag, 1992; 685–695.
18. Song L, Godskesen J. Probabilistic mobility models for mobile and wireless networks. *Proc. the 6th IFIP TC 1/WG 202 international conference on Theoretical Computer Science (TCS'10)*, *IFIP Advances in Informa-*

tion and Communication Technology, vol. 323, Springer Boston, 2010; 86–100.
19. Song L, Godskesen J. Broadcast abstraction in a stochastic calculus for mobile networks. *Proc. the 7th IFIP TC 1/WG 202 international conference on Theoretical Computer Science (TCS'12)*, *lncs*, vol. 7604, Springer-Verlag, 2012; 342–356.
20. Goubault-Larrecq J, Palamidessi C, Troina A. A Probabilistic Applied Pi-Calculus. *Proc. of the 5th Asian Symposium on Programming Languages and Systems (APLAS '07)*, *LNCS*, vol. 4807/2009, Springer-Verlag, 2007; 175–190.
21. Abadi M, Fournet C. Mobile Values, New Names, and Secure Communication. *SIGPLAN Not.* 2001; **36**(3):104–115.
22. Macedonio D, Merro M. A Semantic Analysis of Wireless Network Security Protocols. *NASA Formal Methods*, *lncs*, vol. 7226. Springer Berlin / Heidelberg, 2012; 403–417.
23. Lanotte R, Merro M. Semantic Analysis of Gossip Protocols for Wireless Sensor Networks. *CONCUR 2011 Concurrency Theory*, *lncs*, vol. 6901. Springer Berlin / Heidelberg, 2011; 156–170.
24. Cerone A, Hennessy M. Modelling Probabilistic Wireless Networks. *Logical Methods in Computer Science* 2013; **9**(3):1 – 68.
25. De Nicola R, Katoen JP, Latella D, Massink M. STOKLAIM: A Stochastic Extension of KLAIM. *Technical Report 2006-TR-01*, ISTI 2006.
26. Hillston J. *A Compositional Approach to Performance Modelling*. Distinguished Dissertations in Computer Science, Cambridge University Press, 2005.
27. Gilmore S, Hillston J. The PEPA Workbench: A Tool to Support a Process Algebra-based Approach to Performance Modelling. *Computer Performance Evaluation Modelling Techniques and Tools*, *lncs*, vol. 794. Springer Berlin / Heidelberg, 1994; 353–368.
28. Bernardo M, Bravetti M. Performance Measure Sensitive Congruences for Markovian Process Algebras. *Theoretical Computer Science* 2003; **290**(1):117–160.
29. Mohimani G, Ashtiani F, Javanmard A, Hamdi M. Mobility Modeling, Spatial Traffic Distribution, and Probability of Connectivity for Sparse and Dense Vehicular Ad Hoc Networks. *IEEE Trans. on Vehicular Technology* May, 2009; **58**(4).
30. Beccuti M, Pierro MD, Horvàth A, Horvàth A, Farkas K. A Mean Field Based Methodology for Modeling Mobility in Ad Hoc Networks. *Proc. of 73rd IEEE Vehicular Technology Conference (VTC Spring)*, IEEE: Budapest, HU, 2011; 1–5.
31. Tadayon N, Khoshroo S, Askari E, Wang H, Michel H. Power management in smac-based energy-harvesting wireless sensor networks using queuing analysis. *Journal of Network and Computer Applications* 2013; **36**(3):1008 – 1017.
32. Ross SM. *Stochastic Processes*. 2nd edn., John Wiley & Sons, 1996.
33. Acquaviva A, Aldini A, Bernardo M, Bogliolo A, Bontà E, Lattanzi E. A methodology based on formal methods for predicting the impact of dynamic power management. *Formal Methods for Mobile Computing*, *lncs*, vol. 3465. Springer Berlin / Heidelberg, 2005; 51–58.
34. Perkins C, Bhagwat P. Highly dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers. *Proc. of the Conference on Communications architectures, protocols and applications*, SIGCOMM '94, ACM: New York, NY, USA, 1994; 234–244.
35. Murthy S, Garcia-Luna-Aceves J. An Efficient Routing Protocol for Wireless Networks. *Mobile Networks and Applications* 1996; **1**:183–197.
36. Royer EM, Perkins CE. Multicast Operation of the Ad-hoc On-demand Distance Vector Routing Protocol. *Proc. of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, ACM, 1999; 207–218.
37. Park V, Corson M. A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks. *Proc. of the 16th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '97)*, vol. 3, ieee, 1997; 1405–1413.
38. Johnson DB, Maltz D. Dynamic Source Routing in Ad hoc Wireless Networks. *Mobile Computing*, *The Kluwer International Series in Engineering and Computer Science*, vol. 353. Springer US, 1996; 153–181.
39. Segala R, Lynch N. Probabilistic Simulations for Probabilistic Processes. *Proc. of the 5th International Conference on Concurrency Theory (CONCUR '94)*, *LNCS*, vol. 836, Springer-Verlag, 1994; 481–496.
40. Wattenhofer R, Li L, Bahl P, Wang Y. Distributed Topology Control for Power Efficient Operation in Multihop Wireless Ad hoc Networks. *Proc. 20th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '01).*, vol. 3, ieee, 2001; 1388– 1397.
41. Burkhart M, von Rickenbach P, Wattenhofer R, Zollinger A. Does Topology Control Reduce Interference? *Proc. of the 5th ACM international symposium on Mobile ad hoc networking and computing (MobiHoc '04)*, ACM, 2004; 9–19.
42. Kaplan E. *Understanding GPS: Principles and Applications*. Artech House Publishing, 1996.
43. Tanenbaum AS. *Computer Networks*. Prentice-Hall, 2003.
44. L Gallina TH, Kwiatkowska M, Marin A, Rossi S, Spanò A. Automatic Energy-aware Performance Analysis of Mobile Ad-hoc Networks. *Proc. of IFIP Wireless Days Conference (WD'12)*, IEEE Press, 2012.
45. Le L, Hossain E, Zorzi M. Queueing Analysis for GBN and SR ARQ Protocols under Dynamic Radio Link Adaptation with Non-zero Feedback Delay. *IEEE Transactions on Wireless Communications* 2007; **6**(9):3418–3428.
46. Calamoneri T, Clementi A, Monti A, Rossi G, Silvestri R. Minimum-energy broadcast in random-grid ad-hoc networks: Approximation and distributed algorithms. *Proc. of the 11th International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM '08)*, ACM, 2008; 354–361.
47. Sanchez M, Manzoni P, Haas ZJ. Determination of critical transmission range in ad-hoc networks. *Proc. of the Multiaccess, Mobility and Teletraffic for Wireless Communications Conference (MMT '99)*, 1999.
48. Ferrari G, Malvassori SA, Bragalini M, Tonguz O. Physical layer-constrained routing in ad-hoc wireless networks: a modified aodv protocol with power control. *Proc. of the International Workshop on Wireless Ad-hoc Networks (IWWAN'05)*, 2005.
49. Gomez J, Campbell AT. Variable-range transmission power control in wireless multihop networks. *IEEE Transactions on Mobile Computing (TMC)* 2007; **6**(1):87–99.
50. Hennessy M. A calculus for costed computations. *Logical Methods in Computer Science* 2011; **7**(1).

Appendix

Proof of Theorem 1

1. The first part is proved by induction on the reduction $M \longrightarrow [\![M']\!]_\theta$.
   Let $M \longrightarrow [\![M']\!]_\theta$ due to the application of the rule (R-Move). It means that $M \equiv M' \equiv n[P]_l$, for some name n, location l, some (possibly empty) process P, and $\theta = \mu_l^n$. We simply apply (Move) to obtain:
   $$\frac{}{n[P]_l \xrightarrow{\tau} [\![n[P]_l]\!]_{\mu_l^n}}.$$
   Suppose that $M \longrightarrow [\![M']\!]_\theta$ is due to the application of the rule (R-Par) with $M \equiv M_1 \mid M_2$, $M' \equiv M_1' \mid M_2$ and:
   $$\frac{M_1 \longrightarrow [\![M_1']\!]_\theta}{M_1 \mid M_2 \longrightarrow [\![M_1' \mid M_2]\!]_\theta}.$$
   By induction hypothesis there exist $N \equiv M_1$ and $N' \equiv M_1'$ such that $N \xrightarrow{\tau} [\![N']\!]_\theta$, then by applying rule (Par) we get:
   $$\frac{N \xrightarrow{\tau} [\![N']\!]_\theta}{N \mid M_2 \xrightarrow{\tau} [\![N' \mid M_2]\!]_\theta},$$
   hence by the rules of structural congruence we have that $N \mid M_2 \equiv M_1 \mid M_2 \equiv M$ and $N' \mid M_2 \equiv M_1' \mid M_2 \equiv M'$.
   Suppose that $M \longrightarrow [\![M']\!]_\theta$ is due to the application of the rule (R-Res) with $M \equiv (\nu c)M_1$ and $M' \equiv (\nu c)M_1'$ for some channel c and some networks M_1 and M_1', then
   $$\frac{M_1 \longrightarrow [\![M_1']\!]_\theta}{(\nu c)M_1 \longrightarrow [\![(\nu c)M_1']\!]_\theta}.$$
   By induction hypothesis there exist $N \equiv M_1$ and $N' \equiv M_1'$ such that $N \xrightarrow{\tau} [\![N']\!]_\theta$, then by applying rule (Res), since $\mathtt{Chan}(\tau) \neq c$ we get:
   $$\frac{N \xrightarrow{\tau} [\![N']\!]_\theta}{(\nu c)N \xrightarrow{\tau} [\![(\nu c)N']\!]_\theta},$$
   hence by the rules of structural congruence we have that $(\nu c)N \equiv (\nu c)M_1 \equiv M$ and $(\nu c)N' \equiv (\nu c)M_1' \equiv M'$.
   Let $M \longrightarrow [\![M']\!]_\theta$ due to the application of the rule (R-Bcast). Then $M \equiv n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid \prod_{i\in I} n_i[c(\tilde{x}_i).P_i]_{l_i}$ and $M' \equiv n[P]_l \mid \prod_{i\in I} n_i[P_i\{\tilde{v}/\tilde{x}_i\}]_{l_i}$ for some name n, channel c, location l, radius r, some set L of locations, some tuple $\tilde{v}$ of messages, some (possibly empty) process P, some (possibly empty) set I of networks. By applying the rules (Snd), (Rcv), $\mid I \mid$ times the rule (Bcast) and, finally the rule (Lose), we obtain
   $$n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid \prod_{i\in I} n_i[c(\tilde{x}_i).P_i]_{l_i} \xrightarrow{\tau} [\![n[P]_l \mid \prod_{i\in I} n_i[P_i\{\tilde{v}/\tilde{x}_i\}]_{l_i}]\!]_\Delta .$$
   Finally, suppose that the reduction $M \longrightarrow [\![M']\!]_\theta$ is due to an application of rule (R-Struct):
   $$\frac{M \equiv N \quad N \longrightarrow [\![N']\!]_\theta \quad N' \equiv M'}{M \longrightarrow [\![M']\!]_\theta}.$$
   By induction hypothesis there exist $N_1 \equiv N$ and $N_2 \equiv N'$ such that $N_1 \xrightarrow{\tau} [\![N_2]\!]_\theta$. The statement follows since by applying the rules of the structural congruence we have $M \equiv N \equiv N_1$ and $M' \equiv N' \equiv N_2$.
2. The second part of the theorem follows straightforwardly from Lemma 1 and the definition of Barb.
   $\Rightarrow$ If $M \downarrow_{c@K}$, by the definition of Barb:
   $$M \equiv (\nu\tilde{d})(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid M_1),$$
   for some n, $\tilde{v}$, L, r, some (possibly empty) sequence $\tilde{d}$ with $c \notin \tilde{d}$, some process P and some (possibly empty) network M_1, with $K \subseteq \{k \in L \text{ such that } d(l,k) \leq r\}$ and $K \neq \emptyset$. By applying the rules (Snd), (Par) and (Res):
   $$\frac{n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \xrightarrow{c_L!\tilde{v}[l,r]} [\![n[P]_l]\!]_\Delta}{M \xrightarrow{c_L!\tilde{v}[l,r]} [\![(\nu\tilde{d})(n[P]_l \mid M_1]\!]_\Delta)};$$
   then by rule (Obs): $n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid M_1 \xrightarrow{c!\tilde{v}@K\triangleleft R} [\![n[P]_l \mid M_1]\!]_\Delta$, where $R = \{l' \in Loc : d(l,l') \leq r\}$, and $K \subseteq L \cap R$ as required.
   $\Leftarrow$ If $M \xrightarrow{c!\tilde{v}@K\triangleleft R} [\![M']\!]_\Delta$, because $M \xrightarrow{c_L!\tilde{v}![l,r]} [\![M']\!]_\Delta$, by applying Lemma 1 there exist n, some (possibly empty) sequence $\tilde{d}$ such that $c \notin \tilde{d}$, some process P, some (possibly empty) network M_1 and a set I, such that $\forall i \in I$ with $d(l,l_i) \leq r$:
   $M \equiv (\nu\tilde{d})(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid \prod_{i\in I} n_i[c(\tilde{x}_i).P_i]_{l_i} \mid M_1)$ and $M' \equiv (\nu\tilde{d})(n[P]_l \mid \prod_{i\in I} n_i[P_i\{\tilde{v}/\tilde{x}_i\}]_{l_i} \mid M_1)$.
   Since $K \neq \emptyset$, by the definition of barb we conclude $M \downarrow_{c@K}$.
3. The third part 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), i.e., $M \equiv n[P]_l$, $M' \equiv n[P]_l$, for some name n, some (possibly empty) process P, some location l, $\theta = \mu_l^n$ and
   $$\frac{}{n[P]_l \xrightarrow{\tau} [\![n[P]_l]\!]_{\mu_l^n}},$$
   hence , by applying (R-Move) we get:
   $$\frac{}{n[P]_l \longrightarrow [\![n[P]_l]\!]_{\mu_l^n}}.$$
   If $M \xrightarrow{\tau} [\![M']\!]_\theta$ is due to an application of (Lose):
   $$\frac{M \xrightarrow{c_L!\tilde{v}[l,r]} [\![M']\!]_\Delta}{M \xrightarrow{\tau} [\![M']\!]_\Delta},$$
   for some channel c, some set L of locations, some tuple $\tilde{v}$ of messages, some location l and radius r. By applying Lemma 1, there exist n, $\tilde{v}$, a (possibly empty) sequence $\tilde{d}$ such that $c \notin \tilde{d}$, a process P, a (possibly empty) network M_1 and a (possibly empty) set I with $d(l,l_i) \leq r \ \forall i \in I$ such that:
   $$M \equiv (\nu\tilde{d})(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid \prod_{i\in I} n_i[c(\tilde{x}_i).P_i]_{l_i} \mid M_1)$$
   and
   $$M' \equiv (\nu\tilde{d})(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid \prod_{i\in i} n_i[P_i\{\tilde{v}/\tilde{x}_i\}]_{l_i} \mid M_1).$$

Finally, by applying rules (R-Bcast), (R-Res) and (R-Struct) we get $M \longrightarrow [\![M']\!]_\theta$.
Suppose that $M \xrightarrow{\tau} [\![M']\!]_\theta$ is due to the application of (Res) with $M \equiv (\nu c)M_1$ and $M' \equiv (\nu c)[\![M'_1]\!]_\theta$, for some channel c and for some networks M_1 and M'_1. Then we have:

$$\frac{M_1 \xrightarrow{\tau} [\![M'_1]\!]_\theta}{(\nu c)M_1 \xrightarrow{\tau} [\![(\nu c)M'_1]\!]_\theta}.$$

By induction hypothesis $M_1 \longrightarrow [\![M'_1]\!]_\theta$, hence, by applying rule (R-Res) we get $(\nu c)M_1 \longrightarrow [\![(\nu c)M'_1]\!]_\theta$.
Finally, suppose that $M \xrightarrow{\tau} [\![M']\!]_\theta$ is due to the application of (Par) with $M \equiv M_1 \mid M_2$, $M' \equiv M'_1 \mid M_2$ and

$$\frac{M_1 \xrightarrow{\tau} [\![M'_1]\!]_\theta}{M_1|M_2 \xrightarrow{\tau} [\![M'_1|M_2]\!]_\theta}.$$

By induction hypothesis $M_1 \longrightarrow [\![M'_1]\!]_\theta$, hence, by applying rule (R-Par) we get $M_1|M_2 \longrightarrow [\![M'_1|M_2]\!]_\theta$.

4. The last part of the theorem follows from the definition of barb and Lemma 1. Indeed, 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 1, there exist n, a (possibly empty) sequence $\tilde{d}$ with $c \notin \tilde{d}$, a process P, a (possibly empty) network M_1 and a (possibly empty) set I such that:

$$M \equiv (\nu\tilde{d})(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid \prod_{i\in I} n_i[c(\tilde{x}_i).P_i]_{l_i} \mid M_1)$$

and

$$M' \equiv (\nu\tilde{d})(n[P]_l \mid \prod_{i\in I} n_i[P_i\{\tilde{v}/\tilde{x}_i\}]_{l_i} \mid M_1)\,.$$

Then, by applying the rules (R-Bcast), (R-Par) and (R-Res) we get:
$(\nu\tilde{d})(n[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid \prod_{i\in I} n_i[c(\tilde{x}_i).P_i]_{l_i} \mid M_1)$
$\longrightarrow [\![(\nu\tilde{d})(n[P]_l \mid \prod_{i\in I} n_i[P_i\{\tilde{v}/\tilde{x}_i\}]_{l_i} \mid M_1)]\!]_\Delta$,
and, by applying (R-Struct), we obtain $M \longrightarrow [\![M']\!]_\Delta$, as required. □

Proof of Theorem 2

We have to prove that $\approx_p^{\mathcal{F}}$ is:

1. probabilistic barb preserving
2. reduction closed
3. contextual.

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

Assume that $M \Downarrow_p^F c@K$ for some $F \in \mathcal{F}_\mathcal{C}$. By Definition 3 we have $Prob_M^F(H) = p$, where $H = \{M' : M' \downarrow_{c@K}\}$. We can partition H into a set of equivalence classes with respect to $\approx_p^{\mathcal{F}}$. Formally, $\exists J$ such that $H \subseteq \cup_{j\in J}\mathcal{C}_j$, and $\forall j \in J$ we have $\mathcal{C}_j \in \mathcal{N}/\cong_p^{\mathcal{F}}$ and $H \cap \mathcal{C}_j \neq \emptyset$. Hence:

$$Prob_M^F(H) = \sum_{e\in Exec_M^F(H)} P_M^F(e) = \sum_{j\in J} Prob_M^F(\mathcal{C}_j) = p.$$

By Theorem 1 and by Definition 8 there exists $\hat{F} \in \hat{\mathcal{F}}_\mathcal{C}$ such that $\forall j \in J$:

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

where $\mathcal{C}'_j = \mathcal{C}_j \cup \{\hat{M} \mid \exists \hat{M}' \in \mathcal{C}_j \text{ and } \hat{M} \equiv \hat{M}'\}$.

Now, since $\forall \hat{M}$ such that $\hat{M} \equiv \hat{M}' \in \mathcal{C}_j$, by applying rule (R-Struct) and by Definition 4 $\hat{M} \cong_p^{\mathcal{F}} \hat{M}'$, we obtain that $\{\hat{M} : \hat{M} \equiv \hat{M}' \in \mathcal{C}_j\} \subseteq \mathcal{C}_j$, that means $\mathcal{C}'_j = \mathcal{C}_j$ $\forall j \in J$. Hence we get:

$$\sum_{j\in J} Prob_M^F(\mathcal{C}_j) = \sum_{j\in J} Prob_M^{\hat{F}}(\Longrightarrow, \mathcal{C}_j).$$

Since $M \approx_p^{\mathcal{F}} N$, there exists $\hat{F}' \in \hat{\mathcal{F}}_\mathcal{C}$ such that, by Proposition 2, for all $j \in J$: $Prob_M^{\hat{F}}(\Longrightarrow, \mathcal{C}_j) = Prob_N^{\hat{F}'}(\Longrightarrow, \mathcal{C}_j)$. We then have:

$$p = \sum_{j\in J} Prob_N^{\hat{F}'}(\Longrightarrow, \mathcal{C}_j).$$

Again, by Theorem 1, Proposition 2 and Definition 4, there exists $F' \in \mathcal{F}_\mathcal{C}$ such that for all $j \in J$: $Prob_N^{\hat{F}'}(\Longrightarrow, \mathcal{C}_j) = Prob_N^{F'}(\mathcal{C}_j)$ and

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

i.e., $N \Downarrow_p^{F'} c@K$ as required.

2. To prove that probabilistic labelled bisimilarity $\approx_p^{\mathcal{F}}$ is reduction closed, we have to show that if $M \approx_p^{\mathcal{F}} N$, then for all $F \in \mathcal{F}_\mathcal{C}$, there exists $F' \in \mathcal{F}_\mathcal{C}$ such that for all classes $\mathcal{C} \in \mathcal{N}/\cong_p^{\mathcal{F}}$, $Prob_M^F(\mathcal{C}) = Prob_N^{F'}(\mathcal{C})$.

By Theorem 1 and by Definition 8 we have that $\exists \hat{F} \in \hat{\mathcal{F}}_\mathcal{C}$ such that $Prob_M^F(\mathcal{C}) = Prob_M^{\hat{F}}(\Longrightarrow, \mathcal{C}')$, where $\mathcal{C}' = \mathcal{C} \cup \{\hat{M} : \hat{M} \equiv \hat{M}' \in \mathcal{C}\}$, but since $\forall \hat{M}$ such that $\hat{M} \equiv \hat{M}' \in \mathcal{C}$, by applying rule (R-Struct) and by Definition 4 $\hat{M} \cong_p^{\mathcal{F}} \hat{M}'$ we get $\{\hat{M} : \hat{M} \equiv \hat{M}' \in \mathcal{C}\} \subseteq \mathcal{C}$, i.e., $\mathcal{C}' = \mathcal{C}$.

By Proposition 2 we have that there exists $\hat{F}' \in \hat{\mathcal{F}}_\mathcal{C}$ such that $Prob_M^{\hat{F}}(\Longrightarrow, \mathcal{C}) = Prob_N^{\hat{F}'}(\Longrightarrow, \mathcal{C})$.

Finally, by Theorem 1 and by Definitions 8 and 4, $\exists F' \in \mathcal{F}_\mathcal{C}$ such that $Prob_N^{\hat{F}'}(\Longrightarrow, \mathcal{C}) = Prob_N^{F'}(\mathcal{C})$, as required.

3. In order to prove that probabilistic labelled bisimilarity $\approx_p^{\mathcal{F}}$ is contextual we have to prove that, if $M \approx_p^{\mathcal{F}} N$:

1. $M \mid O \approx_p^{\mathcal{F}} N \mid O$ $\forall O \in \mathcal{N}$.
2. $(\nu d)M \approx_p^{\mathcal{F}} (\nu d)N$ $\forall d \in \mathbf{C}$.

Case 1. Let us consider the relation

$$\mathcal{R} = \{(M \mid O, N \mid O) : M \approx_p^{\mathcal{F}} N\}.$$

We prove that for all scheduler $F \in \hat{\mathcal{F}}_\mathcal{C}$ there exists a scheduler $F' \in \hat{\mathcal{F}}_\mathcal{C}$ such that for all α and for all classes $\mathcal{C}$ in $\mathcal{N}/\approx_p^{\mathcal{F}}$:

1. if $\alpha = \tau$ then $Prob_{M|O}^F(\xrightarrow{\tau}, \mathcal{C}) = Prob_{N|O}^{F'}(\Longrightarrow, \mathcal{C})$.
   Indeed, 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 $\bar{P} \approx_p^{\mathcal{F}} \bar{Q}$. 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 $M \mid O \xrightarrow{\tau} [\![M \mid O']\!]_\theta$ because $O \xrightarrow{\tau} [\![O']\!]_\theta$ the proof is simple, because for all $\bar{M}$ in the support of $[\![M \mid O']\!]_\theta$ such that $\bar{M} \in \mathcal{C}$, it holds $\bar{M} \equiv M \mid O''$ and, since $M \approx_p^{\mathcal{F}} N$, $N \mid O'' \in \mathcal{C}$ too, by definition of $\mathcal{R}$. By Definition 4 there exists $\bar{F} \in \mathcal{F}_\mathcal{C}$ such that, by applying rule (R-Par) to the reduction $O \longrightarrow [\![O']\!]_\theta$,

$N \mid O \longrightarrow [\![O' \mid N]\!]_\theta \in Exec^{\bar{F}}_{N|O}$. By Theorem 1 and by Definition 8 $\exists F' \in \hat{\mathcal{F}}_\mathcal{C}$ such that $Prob^{\bar{F}}_{N|O}(\mathcal{C}) = Prob^{F'}_{N|O}(\Longrightarrow, \mathcal{C})$, hence we have $Prob^{F}_{M|O}(\xrightarrow{\tau}, \mathcal{C}) = Prob^{F'}_{N|O}(\Longrightarrow, \mathcal{C})$ as required.

(ii) If $M \mid O \xrightarrow{\tau} [\![M' \mid O]\!]_\theta$ because $M \xrightarrow{\tau} [\![M']\!]_\theta$, then by Definition 8 there exists $F_1 \in \hat{\mathcal{F}}_\mathcal{C}$ such that $Prob^{F}_{M|O}(\xrightarrow{\tau}, \mathcal{C}) = Prob^{F_1}_{M}(\xrightarrow{\tau}, \mathcal{D})$. Since $M \approx^{\mathcal{F}}_p N$, there exists $F_2 \in \hat{\mathcal{F}}_\mathcal{C}$ such that $Prob^{F_1}_{M}(\xrightarrow{\tau}, \mathcal{D}) = Prob^{F_2}_{N}(\Longrightarrow, \mathcal{D})$. For each $N \xrightarrow{\tau}_{\theta_1} \ldots \xrightarrow{\tau}_{\theta_k} N_k \in Exec^{F_1}_{N}(\Longrightarrow, \mathcal{D})$, there exists a scheduler $\bar{F} \in \mathcal{F}_\mathcal{C}$ such that $N \longrightarrow_{\theta_1} N_1 \ldots \longrightarrow_{\theta_k} N_k \in Exec^{\bar{F}}_{N}$. By Definition 4, since $\mathcal{F}_\mathcal{C}$ captures the interactions of N with any context, $\exists \bar{F}' \in \mathcal{F}_\mathcal{C}$ such that, by applying rule (R-Par) to each step in e: $N \mid O \longrightarrow_{\theta_1} \ldots \longrightarrow_{\theta_k} N_k \mid O \in Exec^{\bar{F}'}_{N|O}$. By Definition 8 we finally get $F' \in \hat{\mathcal{F}}_\mathcal{C}$ such that:

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

(iii) If $M \mid O \xrightarrow{\tau} [\![M' \mid O']\!]_\Delta$ due to a synchronization between M and O, then there are two cases to consider. If $M \xrightarrow{c_L!\tilde{v}[l,r]} [\![M']\!]_\Delta$ and $O \xrightarrow{c?\tilde{v}@k} [\![O']\!]_\Delta$, for some tuple $\tilde{v}$ of messages, channel c, locations l, k and radius r, such that $d(l,k) \leq r$, we can apply rule (Obs) obtaining $M \xrightarrow{c!\tilde{v}@K \triangleleft R} [\![M']\!]_\Delta$ for some set $R = \{l' \mid d(l,l') \leq r\}$ with $k \in R$ and $K = L \cap R$.
Hence, by Definition 8, there exists $F_1 \in \hat{\mathcal{F}}_\mathcal{C}$ such that $Prob^{F}_{M|O}(\xrightarrow{\tau}, \mathcal{C}) = Prob^{F_1}_{M}(\xrightarrow{c!\tilde{v}@K \triangleleft R}, \mathcal{D})$. Moreover, since $N \approx^{\mathcal{F}}_p M$, there exists $F_2 \in \hat{\mathcal{F}}_\mathcal{C}$ such that $Prob^{F_1}_{M}(\xrightarrow{c!\tilde{v}@K \triangleleft R}, \mathcal{D}) = Prob^{F_2}_{N}(\overset{c!\tilde{v}@K \triangleleft R}{\Longrightarrow}, \mathcal{D})$, where each execution $e \in Exec^{F_2}_{N}(\overset{c!\tilde{v}@K \triangleleft R}{\Longrightarrow}, \mathcal{D})$ has the form

$$e = N \xrightarrow{\tau}_{\theta_1} N_1 \xrightarrow{\tau}_{\theta_2} \ldots N_{i-1} \xrightarrow{c!\tilde{v}@K \triangleleft R}_\Delta N_i \xrightarrow{\tau}_{\theta_{i+1}} \ldots N',$$

with $k \in R$, 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 the probability. Finally if we take $F' \in LSched$ which applies rule (Lose) to the output action, we obtain the required result:

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

We have finally to prove that $F' \in \hat{\mathcal{F}}_\mathcal{C}$. We start by the consideration that, by Definition 1, for any execution of the form $\overset{\alpha}{\Longrightarrow}$ in $\hat{\mathcal{F}}_\mathcal{C}$, where α is a silent or an output action there exists a correspondent reduction in $\mathcal{F}_\mathcal{C}$. Since by Definition 4, for any context, there exists a scheduler in $\mathcal{F}_\mathcal{C}$ mimicking the behaviour exhibited by N when interacting with the given context, we can affirm that $\exists \bar{F} \in \mathcal{F}_\mathcal{C}$ such that $Exec^{\bar{F}}_{N|O}$ contains all the reductions corresponding to the executions of $Exec^{F'}_{N|O}$. Hence, by Definition 8, $F' \in \hat{\mathcal{F}}_\mathcal{C}$, 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 $\tilde{v}$, channel c, locations l, k and radius r, such that $d(l,k) \leq r$, then by Definition 8 $\exists F_1 \in \hat{\mathcal{F}}_\mathcal{C}$ such that:

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

and, since $M \approx^{\mathcal{F}}_p N$, there exists $F_2 \in \hat{\mathcal{F}}_\mathcal{C}$ such that $Prob^{F_1}_{M}(\xrightarrow{c?\tilde{v}@k}, \mathcal{D}) = Prob^{F_2}_{N}(\overset{c?\tilde{v}@k}{\Longrightarrow}, \mathcal{D})$ 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 $k \in R$, also N is able to synchronize with O, for all executions in $Exec^{F_2}_{N}(\overset{c?\tilde{v}@k}{\Longrightarrow}, \mathcal{D})$ of the form $e = N \xrightarrow{\tau}_{\theta_1} N_1 \xrightarrow{\tau}_{\theta_2} \ldots N_{i-1} \xrightarrow{c?\tilde{v}@k}_\Delta N_i \xrightarrow{\tau}_{\theta_{i+1}} \ldots N'$ since by hypothesis $d(l,k) \leq r$, then by applying rule (Bcast) we get $N_{i-1} \mid O \xrightarrow{c_L!\tilde{v}[l.r]} N_i \mid O'$, and there exists a matching execution: $N \mid O \xrightarrow{\tau}_{\theta_1} N_1 \mid O \xrightarrow{\tau}_{\theta_2} \ldots N_{i-1} \mid O \xrightarrow{c_L!\tilde{v}[l,r]}_\Delta N_i \mid O' \xrightarrow{\tau}_{\theta_{i+1}} \ldots N' \mid O'$.
By rule (Lose) to $N_{i-1} \mid O \xrightarrow{c_L!\tilde{v}[l,r]}_\Delta N_i \mid O'$ and by Definition 4 $\exists \bar{F}' \in \mathcal{F}_\mathcal{C}$ such that, $Prob^{\bar{F}'}_{N|O}(\mathcal{C}) = Prob^{F_2}_{N}(\mathcal{D})$. By Definition 8 $\exists F' \in \hat{\mathcal{F}}_\mathcal{C}$ such that, $Prob^{F'}_{N|O}(\Longrightarrow, \mathcal{C}) = Prob^{\bar{F}'}_{N|O}(\mathcal{C})$. If N is not able to receive the message the proof is analogous, because $\exists F' \in \hat{\mathcal{F}}_\mathcal{C}$ such that, for each execution in $Exec^{F_1}_{N}(\Longrightarrow, \mathcal{D})$ of the form $N \xrightarrow{\tau}_{\theta_1} N_1 \ldots \xrightarrow{\tau}_{\theta_k} N_k$, by applying rule (Par) to each step we have that $N \mid O \xrightarrow{\tau}_{\theta_1} N_1 \mid O \ldots \xrightarrow{\tau}_{\theta_k} N_k \mid O$, and by applying rule (Bcast) and (Lose) to O, and then (Par) to $N_k \mid O$, we get:
$N \mid O \xrightarrow{\tau}_{\theta_1} N_1 \mid O \ldots \xrightarrow{\tau}_{\theta_k} N_k \mid O \xrightarrow{\tau}_\Delta N_k \mid O' \in Exec^{F'}_{N|O}(\Longrightarrow, \mathcal{C})$, hence, since the output of O does not change the probabilities of the executions, we get: $Prob^{F}_{M|O}(\Longrightarrow, \mathcal{C}) = Prob^{F_1}_{M}(\Longrightarrow, \mathcal{D}) = Prob^{F_2}_{N}(\Longrightarrow, \mathcal{D}) = Prob^{F'}_{N|O}(\Longrightarrow, \mathcal{C})$.

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

$$Prob^{F}_{M|O}(\xrightarrow{c!\tilde{v}@K \triangleleft R}, \mathcal{C}) = Prob^{F'}_{N|O}(\overset{c!\tilde{v}@K \triangleleft R}{\Longrightarrow}, \mathcal{C}).$$

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

3. if $\alpha = c?\tilde{v}@k$ then it holds

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

or $Prob^{F}_{M|O}(\xrightarrow{\alpha}, \mathcal{C}) = Prob^{F'}_{N|O}(\Longrightarrow, \mathcal{C})$. If $P, Q \in \mathcal{C}$, then by definition of $\mathcal{R}$, $P \equiv \bar{P} \mid \bar{O}$, $Q \equiv \bar{Q} \mid \bar{O}$ and $\bar{P} \approx^{\mathcal{F}}_p \bar{Q}$. Hence there exists $\mathcal{D} \in \mathcal{N}/\approx^{\mathcal{F}}_p$ such that $\mathcal{D} = \{\bar{P} : \bar{P} \mid \bar{O} \in \mathcal{C}\}$. 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 $M \approx^{\mathcal{F}}_p N$, then also $N \mid O' \in \mathcal{C}$, and, by Definition 8 there exists $F' \in \hat{\mathcal{F}}_\mathcal{C}$ such that by applying rule (Par) to $O \xrightarrow{\alpha} O'$, we get $N \mid O \xrightarrow{\alpha} N \mid O'$ obtaining:

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

(ii) The transition is due to an action performed by M. By Definition 8 $\exists F_1 \in \hat{\mathcal{F}}_\mathcal{C}$ such that $Prob^{F}_{M|O}(\xrightarrow{\alpha}, \mathcal{C}) = Prob^{F_1}_{M}(\xrightarrow{\alpha}, \mathcal{D})$. Since $M \approx^{\mathcal{F}}_p N$, there exists $F_2 \in \hat{\mathcal{F}}_\mathcal{C}$ such that $Prob^{F_1}_{M}(\xrightarrow{\alpha}, \mathcal{D}) = Prob^{F_2}_{N}(\overset{\alpha}{\Longrightarrow}, \mathcal{D})$, or $Prob^{F_1}_{M}(\xrightarrow{\alpha}, \mathcal{D}) = Prob^{F_2}_{N}(\Longrightarrow, \mathcal{D})$. In both cases,

for $e \in Exec_N^{F_1}(\stackrel{\hat{\alpha}}{\Longrightarrow}, \mathcal{D})$: $e = N \xrightarrow{\alpha_1}_{\theta_1} N_1 ... \xrightarrow{\alpha_k}_{\theta_k} N_k$ by rule (Par) to each step we get:
$N \mid O \xrightarrow{\alpha_1}_{\theta_1} N_1 \mid O ... \xrightarrow{\alpha_k}_{\theta_k} N_k \mid O$.
Then, we have that $\exists F' \in LSched$ such that

$$Prob_N^{F_2}(\stackrel{\alpha}{\Longrightarrow}, \mathcal{D}) = Prob_{N\mid O}^{F'}(\stackrel{\alpha}{\Longrightarrow}, \mathcal{C}),$$

or

$$Prob_N^{F_2}(\Longrightarrow, \mathcal{D}) = Prob_{N\mid O}^{F'}(\Longrightarrow, \mathcal{C}).$$

In order to prove that $F' \in \hat{\mathcal{F}}_{\mathcal{C}}$, we start by the consideration that, by Definition 8 there exists at least a context $C[\cdot]$ and $\exists \bar{F} \in \mathcal{F}_{\mathcal{C}}$ such that $C[N] \longrightarrow C'[N']$, and, by the reduction rules we get:

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

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 4 there exists a scheduler allowing $m[\bar{c}_{L,r}\langle \tilde{v}\rangle.P]_l \longrightarrow [\![m[P]_l]\!]_{\Delta}$, and again by Definition 4 there exists a scheduler such that $m[\bar{c}_{L,r}\langle \tilde{v}\rangle.P]_l \mid N \mid O \longrightarrow^* [\![m[P]_l \mid N' \mid O']\!]_{\Delta}$, and hence, by Definition 8, $F' \in \hat{\mathcal{F}}_{\mathcal{C}}$ as required.

Case 2. Let us consider now the relation

$$\mathcal{S} = \{((\nu d)M, (\nu d)N) : M \approx_p^{\mathcal{F}} N\}.$$

Let $\mathcal{C} \in \mathcal{N}/\mathcal{S}$: if $P, Q \in \mathcal{C}$, then by definition of $\mathcal{S}$ we have $P \equiv (\nu \bar{d})\bar{P}$, $Q \equiv (\nu \bar{d})\bar{Q}$ and $\bar{P} \approx_p^{\mathcal{F}} \bar{Q}$. Hence $\exists \mathcal{D} \in \mathcal{N}/\approx_p^{\mathcal{F}}$ such that $\mathcal{D} = \{\bar{P} : (\nu \bar{d})\bar{P} \in \mathcal{C}\}$.

We have to prove that, $\forall F \in \hat{\mathcal{F}}_{\mathcal{C}}$, $\exists F' \in \hat{\mathcal{F}}_{\mathcal{C}}$ such that, $\forall \mathcal{C} \in \mathcal{N}/\mathcal{S}$, $\forall \alpha$:

1. $\alpha = \tau$ implies $Prob_{(\nu d)M}^{F}(\xrightarrow{\tau}, \mathcal{C}) = Prob_{(\nu d)N}^{F'}(\Longrightarrow, \mathcal{C})$. Since $\mathtt{Chan}(\tau) = \perp$, by Definition 8 $\exists F_1 \in \hat{\mathcal{F}}_{\mathcal{C}}$ such that $Prob_{(\nu d)M}^{F}(\xrightarrow{\tau}, \mathcal{C}) = Prob_{M}^{F_1}(\xrightarrow{\tau}, \mathcal{D})$ and, since $M \approx_p^{\mathcal{F}} N$ $\exists F_2 \in \hat{\mathcal{F}}_{\mathcal{C}}$ 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, we have

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

   In order to prove that $F' \in \hat{\mathcal{F}}_{\mathcal{C}}$, we start by the consideration that, by Definition 4, for any context there exists a scheduler in $\mathcal{F}_{\mathcal{C}}$ mimicking the behaviour of N when interacting with the given context. Hence $\exists \bar{F} \in \mathcal{F}_{\mathcal{C}}$ such that $Exec_{(\nu d)N}^{\bar{F}}$ contains all the reductions corresponding to the executions in $Exec_{(\nu d)N}^{F'}$, i.e., by Definition 8, $F' \in \hat{\mathcal{F}}_{\mathcal{C}}$.
2. $\alpha = c!\tilde{v}@K \triangleleft R$. Since $\mathtt{Chan}(c!\tilde{v}@K \triangleleft R) \neq d$, by Definition 8 $\exists F_1 \in \hat{\mathcal{F}}_{\mathcal{C}}$ with $Prob_{(\nu d)M}^{F}(\xrightarrow{\alpha}, \mathcal{C}) = Prob_{M}^{F_1}(\xrightarrow{\alpha}, \mathcal{D})$. Since $M \approx_p^{\mathcal{F}} N$, $\exists F_2 \in \hat{\mathcal{F}}_{\mathcal{C}}$ such that $Prob_{M}^{F_1}(\xrightarrow{\alpha}, \mathcal{D}) = Prob_{N}^{F'}(\stackrel{\alpha}{\Longrightarrow}, \mathcal{D})$. Since $\mathtt{Chan}(\alpha) \neq d$, $\exists F' \in LSched$ with $Prob_{N}^{F_2}(\stackrel{\alpha}{\Longrightarrow}, \mathcal{D}) = Prob_{(\nu d)N}^{F_2}(\stackrel{\alpha}{\Longrightarrow}, \mathcal{C})$. We now can prove that $F' \in \hat{\mathcal{F}}_{\mathcal{C}}$ as in the previous cases.
3. $\alpha = c?\tilde{v}@k$. Again, by $\mathtt{Chan}(c?\tilde{v}@k) \neq d$, by Definition 8 $\exists F_1 \in \hat{\mathcal{F}}_{\mathcal{C}}$ with $Prob_{(\nu d)M}^{F}(\xrightarrow{\alpha}, \mathcal{C}) = Prob_{M}^{F_1}(\xrightarrow{\alpha}, \mathcal{D})$. Since $M \approx_p^{\mathcal{F}} N$, $\exists F_2 \in \hat{\mathcal{F}}_{\mathcal{C}}$ such that $Prob_{M}^{F_1}(\xrightarrow{\alpha}, \mathcal{D}) = Prob_{N}^{F_2}(\stackrel{\alpha}{\Longrightarrow}, \mathcal{D})$ or $Prob_{M}^{F_1}(\xrightarrow{\alpha}, \mathcal{D}) = Prob_{N}^{F_2}(\Longrightarrow, \mathcal{D})$, in the case that N is not able to receive $\tilde{v}$. In both cases, by rule (Res) to N, since $\mathtt{Chan}(\tau) = \perp$ and $\mathtt{Chan}(c?\tilde{v}@k) \neq d$. Hence, $\exists F' \in LSched$ such that

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

   or

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

   Again, we prove that $F' \in \hat{\mathcal{F}}_{\mathcal{C}}$ as in the previous cases. □

Proof of Theorem 3

In order to prove the completeness of the probabilistic labelled bisimilarity 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, $\forall F \in \hat{\mathcal{F}}_{\mathcal{C}}$ $\exists F' \in \hat{\mathcal{F}}_{\mathcal{C}}$ such that, $\forall \mathcal{C} \in \mathcal{N}/\mathcal{R}$, $\forall \alpha$:

if $\alpha = \tau$ then $Prob_M^F(\xrightarrow{\tau}, \mathcal{C}) = Prob_N^{F'}(\Longrightarrow, \mathcal{C})$.
By Theorem 1 and Definition 8 we know that $\exists \bar{F} \in \mathcal{F}_{\mathcal{C}}$ such that $Prob_M^F(\xrightarrow{\tau}, \mathcal{C}) = Prob_M^{\bar{F}}(\mathcal{C})$. By $M \cong_p^{\mathcal{F}} N$, $\exists \bar{F}' \in \mathcal{F}_{\mathcal{C}}$ such that $Prob_M^{\bar{F}}(\mathcal{C}) = Prob_N^{\bar{F}'}(\mathcal{C})$. Again by Theorem 1 and by Definition 8 $\exists F' \in \hat{\mathcal{F}}_{\mathcal{C}}$ such that $Prob_N^{\hat{F}'}(\mathcal{C}) = Prob_N^{F'}(\Longrightarrow, \mathcal{C} \cup \{\bar{N} \equiv N' \in \mathcal{C}\})$, but since $\cong_p^{\mathcal{F}}$ is closed under structural equivalence, $\forall \bar{N} \equiv N' \in \mathcal{C}$, $\bar{N} \in \mathcal{C}$, and hence: $Prob_M^F(\xrightarrow{\tau}, \mathcal{C}) = Prob_N^{F'}(\Longrightarrow, \mathcal{C})$.
if $\alpha = c!\tilde{v}@K \triangleleft R$ then $Prob_M^F(\xrightarrow{\alpha}, \mathcal{C}) = Prob_N^{F'}(\stackrel{\alpha}{\Longrightarrow}, \mathcal{C})$.
First note that $Prob_M^F(\xrightarrow{c!\tilde{v}@K \triangleleft R}, \mathcal{C})$ is either 0 or 1.
If $Prob_M^F(\xrightarrow{c!\tilde{v}@K \triangleleft R}, \mathcal{C}) = 0$ we are done, because it will be enough to take any scheduler $F' \in \hat{\mathcal{F}}_{\mathcal{C}}$ not allowing observable output actions on the channel c, and we get $Prob_M^F(\xrightarrow{c!\tilde{v}@K \triangleleft R}, \mathcal{C}) = Prob_N^{F'}(\stackrel{c!\tilde{v}@K \triangleleft R}{\Longrightarrow}, \mathcal{C})$.
If $Prob_M^F(\xrightarrow{c!\tilde{v}@K \triangleleft R}, \mathcal{C}) = 1$, by Theorem 1 and by Definition 8 $\exists \bar{F} \in \mathcal{F}_{\mathcal{C}}$ such that $M \Downarrow_1^{\bar{F}} c@K$, and this means that $\exists \bar{F}' \in \mathcal{F}_{\mathcal{C}}$ such that $N \Downarrow_1^{\bar{F}'} c@K$, hence, by Theorem 1 and by Definition 8 there exist $F' \in \hat{\mathcal{F}}_{\mathcal{C}}$ and R' such that $K \subseteq R'$ and $Prob_N^{\bar{F}'}(\mathcal{C}) = Prob_N^{F'}(\stackrel{c!\tilde{v}@K \triangleleft R'}{\Longrightarrow}, \mathcal{C})$.
We proved that $\exists R'$ with

$$Prob_M^F(\xrightarrow{c!\tilde{v}@K \triangleleft R}, \mathcal{C}) = Prob_N^{F'}(\stackrel{c!\tilde{v}@K \triangleleft R'}{\Longrightarrow}, \mathcal{C}),$$

now we want to show that $R' = R$. In order to mimic the effect of the action $c!\tilde{v}@K \triangleleft R$, we build the following context

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

where $R = \{k_1, ..., k_n\}$, n_i, m_i, $\mathtt{ok}^{(i)}$ and $\mathtt{f}^{(i)}$ are fresh $\forall i \in [1-n]$. Since $M \xrightarrow{c!\tilde{v}@K \triangleleft R}$, then the message is reachable by all nodes n_i, hence, by Definition 4 $\exists \bar{F}_1 \in \mathcal{F}_{\mathcal{C}}$ such that $C[M] \longrightarrow^* \hat{M}$, where

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

with $\hat{M} \not\downarrow_{\mathtt{f}^{(i)}@R}$ and $\hat{M}\Downarrow_1^{\bar{F}_1}\mathtt{ok}^{(i)}@R$, $\forall i \in [1-n]$.
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 $C[M] \cong_p^{\mathcal{F}} C[N]$, $\exists \bar{F}_2 \in \mathcal{F}_{\mathcal{C}}$ such that $Prob^{\bar{F}_1}_{C[M]}(\mathcal{C}') = Prob^{\bar{F}_2}_{C[N]}(\mathcal{C}')$ where $\hat{M} \in \mathcal{C}'$.
Therefore, $C[N] \longrightarrow^* \hat{N}$ with $\hat{N} \not\downarrow_{\mathtt{f}^{(i)}@R}$ and $\hat{N}\Downarrow_1^{\bar{F}_2}\mathtt{ok}^{(i)}@R$.
The constraints on the barbs allow us to deduce that

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

which implies $N \stackrel{c!\tilde{v}@K\triangleleft R}{\Longrightarrow} N'$, or $N \Longrightarrow N'$ in case (Lose) has been applied to the output action on the channel c. Since $\hat{M}, \hat{N} \in \mathcal{C}$, then $\hat{M} \cong_p^{\mathcal{F}} \hat{N}$, and since $\cong_p^{\mathcal{F}}$ is contextual, it results $(\nu\mathtt{ok}^{(1)}...\mathtt{ok}^{(n)})\hat{M} \cong_p^{\mathcal{F}_{\mathcal{M}}} (\nu\mathtt{ok}^{(1)}...\mathtt{ok}^{(n)})\hat{N}$.
By applying (Struct Res Par):

$$(\nu\mathtt{ok}^{(1)}...\mathtt{ok}^{(n)})\hat{M} \equiv M' \mid (\nu\mathtt{ok}^{(1)}...\mathtt{ok}^{(n)})\prod_{i=1}^{n}(m_i[\bar{\mathtt{ok}}^{(i)}_{k_i,r}\langle\tilde{v_i}\rangle]_{k_i}) \equiv M'$$

and

$$(\nu\mathtt{ok}^{(1)}...\mathtt{ok}^{(n)})\hat{N} \equiv N' \mid (\nu\mathtt{ok}^{(1)}...\mathtt{ok}^{(n)})\prod_{i=1}^{n}(m_i[\bar{\mathtt{ok}}^{(i)}_{k_i,r}\langle\tilde{v_i}\rangle]_{k_i}) \equiv N'$$

and, since the network

$$(\nu\mathtt{ok}^{(1)}...\mathtt{ok}^{(n)})\prod_{i=1}^{n}(m_i[\bar{\mathtt{ok}}^{(i)}_{k_i,r}\langle\tilde{v_i}\rangle]_{k_i})$$

is silent, we can derive $M' \cong_p^{\mathcal{F}} N'$. Since $N' \in \mathcal{C}$ and $N \stackrel{c!\tilde{v}@K\triangleleft R}{\Longrightarrow} N'$, by Definition 8 $\exists F' \in \hat{\mathcal{F}}_{\mathcal{C}}$ such that $Prob^{F'}_N(\stackrel{c!\tilde{v}@K\triangleleft R}{\Longrightarrow}, \mathcal{C}) = 1 = Prob^{F}_M(\stackrel{c!\tilde{v}@K\triangleleft R}{\Longrightarrow}, \mathcal{C})$, as required.

if $\alpha = c?\tilde{v}@k$ then $Prob^{F}_M(\xrightarrow{\alpha}, \mathcal{C}) = Prob^{F'}_N(\stackrel{\alpha}{\Longrightarrow}, \mathcal{C})$ or $Prob^{F'}_N(\Longrightarrow, \mathcal{C})$.
We notice that $Prob^{F}_M(\xrightarrow{c?\tilde{v}@k}, \mathcal{C})$ is either 0 or 1. If $Prob^{F}_M(\xrightarrow{c?\tilde{v}@k}, \mathcal{C}) = 0$ we are done, because it will be enough to take any scheduler $F' \in \hat{\mathcal{F}}_{\mathcal{C}}$ not allowing input actions on the channel c. Therefore we obtain that $Prob^{F}_M(\xrightarrow{c?\tilde{v}@k}, \mathcal{C}) = Prob^{F'}_N(\stackrel{c?\tilde{v}@k}{\Longrightarrow}, \mathcal{C})$.
If $Prob^{F}_M(\xrightarrow{c?\tilde{v}@k}, \mathcal{C}) = 1$, because $M \xrightarrow{c?\tilde{v}@k} [\![M']\!]_{\Delta}$, by Definition 4 there exists at least a context $C[\cdot]$ and $\exists \bar{F} \in \mathcal{F}_{\mathcal{C}}$ such that $C[M] \longrightarrow C'[M']$, and by Theorem 1 we have $C[\cdot] \equiv (\nu\tilde{d})m[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid M_1$ and $C'[\cdot] \equiv (\nu\tilde{d})m[P]_l \mid M_1'$ for some m, some tuple $\tilde{d}$ of channels such that $c \notin \tilde{d}$, some set L of messages, some radius r, some process P, some location l such that $d(l,k) \leq r$ and some (possibly empty) networks M_1 and M_1'. By Definition 4, for any context there exists a scheduler in $\mathcal{F}_{\mathcal{C}}$ allowing m to perform the output when interacting with any context.
Hence we can build the following context:
$C_1[\cdot] = \cdot \mid m[\bar{c}_{L,r}\langle\tilde{v}\rangle.P]_l \mid m_1[c(\tilde{x}).\bar{\mathtt{f}}_{k,r'}\langle\tilde{x}\rangle.\bar{\mathtt{ok}}_{k,r'}\langle\tilde{x}\rangle]_k$,
in order to mimic the behaviour of the networks, with m static, $\mathtt{f}$ and $\mathtt{ok}$ fresh channels, $r' > 0$ and $d(l,k) > r'$ $\forall l \in Loc$ such that $l \neq k$. Hence, $\exists \bar{F}_1 \in \mathcal{F}_{\mathcal{C}}$ such that $C_1[M] \longrightarrow^* M' \mid m[P]_l \mid m_1[\bar{\mathtt{ok}}_{k,r'}\langle\tilde{v}\rangle]_k \in Exec^{\bar{F}_1}_{C[M]}$, with $M' \mid m[P]_l \mid m[\bar{\mathtt{ok}}_{k,r'}\langle\tilde{v}\rangle]_k \not\downarrow_{\mathtt{f}@k}$ and $M' \mid m[P]_l \mid m[\bar{\mathtt{ok}}_{k,r'}\langle\tilde{v}\rangle]_k\Downarrow_1^{\bar{F}_1}\mathtt{ok}@k$.
The reduction sequence above must be matched by a corresponding reduction sequence of the form
$C_1[N] \longrightarrow^* N' \mid m[P]_l \mid m[\bar{\mathtt{ok}}_{k,r'}\langle\tilde{v}\rangle]_k$, with

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

and
$N' \mid m[P]_l \mid m[\bar{\mathtt{ok}}_{k,r'}\langle\tilde{v}\rangle]_k\Downarrow_1^{\bar{F}_2}\mathtt{ok}@k$ for some $\bar{F}_2 \in \mathcal{F}_{\mathcal{C}}$.
This does not ensure that N actually performed the input action, but we can conclude that $\exists F' \in LSched$ and N' such that either $N \stackrel{c?\tilde{v}@k}{\Longrightarrow} N'$ or $N \Longrightarrow N'$. Since $M' \mid m[P]_l \mid m[\bar{\mathtt{ok}}_{k,r'}\langle\tilde{v}\rangle]_k \cong_p N' \mid m[P]_l \mid m[\bar{\mathtt{ok}}_{k,r'}\langle\tilde{v}\rangle]_k$ and $\cong_p^{\mathcal{F}}$ is is a contextual relation, we can easily derive $M' \cong_p^{\mathcal{F}} N'$ (applying the rules for structural equivalence), i.e., there exists $F' \in LSched$ such that:
$Prob^{F}_M(\xrightarrow{c?\tilde{v}@k}, \mathcal{C}) = 1 = Prob^{F'}_N(\stackrel{c?\tilde{v}@k}{\Longrightarrow}, \mathcal{C})$ or
$Prob^{F}_M(\xrightarrow{c?\tilde{v}@k}, \mathcal{C}) = 1 = Prob^{F'}_N(\Longrightarrow, \mathcal{C})$.
Now we have only to prove that $F' \in \hat{\mathcal{F}}_{\mathcal{C}}$, but this follows straightforwardly by Definition 8, since $\bar{F}_2 \in \mathcal{F}_{\mathcal{C}}$. □