An Observational Theory for Mobile Ad Hoc Networks

Massimo Merro
Department of Computer Science, University of Verona, Italy
e-mail: Massimo.Merro@univr.it

Questo rapporto è disponibile su Web all’indirizzo:
This report is available on the web at the address:
http://www.di.univr.it/report
Abstract

We propose a process calculus to study the observational theory of Mobile Ad Hoc Networks. The operational semantics of our calculus is given both in terms of a Reduction Semantics and in terms of a Labelled Transition Semantics. We prove that the two semantics coincide. The labelled transition system is then used to derive the notions of simulation and bisimulation for ad hoc networks. As a main result, we prove that the (weak) labelled bisimilarity completely characterises (weak) reduction barbed congruence, a standard, branching-time, contextually-defined program equivalence. We then use our (bi)simulation proof method to formally prove a number of non-trivial properties of ad hoc networks.
1 Introduction

Wireless technology has exploded in popularity in the last years. Its applications span from user applications such as personal area networks, ambient intelligence, and wireless local area networks, to real-time applications, such as cellular and ad hoc networks.

Ad hoc networking is a new area in wireless communications that is attracting the attention of many researchers, for its potential to provide ubiquitous connectivity without the assistance of any fixed infrastructure. A Mobile Ad Hoc Network (MANET) is an autonomous system composed of both stationary and mobile devices communicating with each other via radio transceivers. Mobile devices are free to move randomly and organise themselves arbitrarily; thus, the network’s wireless topology may change rapidly and unpredictably. Stationary devices cannot move i.e. their physical location does not vary with time. The network may operate in a standalone fashion, or may be connected to the larger Internet. MANETs can be used wherever a wired backbone is infeasible and/or economically inconvenient, for example, to provide communications during emergencies, special events (expos, concerts, etc.), or in hostile environments.

Wireless devices use radio frequency channels to broadcast messages to the other devices. However, this form of broadcast is quite different from the more conventional wired-based broadcast that we find in networks with Ethernet and that, from a semantic point of view, is well-understood [20, 21, 8]. First, in Ethernet-like systems broadcasting is global, i.e., the messages transmitted reach all nodes of the system. By contrast, in wireless system broadcasting is local, i.e., a transmission spans over a limited area, called cell, and therefore reaches only a -possibly empty- subset of the devices in the system. Actually, even the devices within a cell might not be reachable due to environmental conditions such as walls, obstacles, etc. Second, in wireless systems channels are half-duplex: on a given channel, a device can either transmit or receive, but cannot do both at the same time. Hence, an interference between two transmissions is only possibly detected by receivers located in the intersection of the cells of the two transmitters. Interference is thus a delicate aspect of wireless systems that is handled by means of specific protocols (e.g., CSMA/CA).

In mobile ad hoc networks there is a further catch: the set of nodes that lie within the cell of a node can change unpredictably due to node movement or node failure, thereby altering the set of nodes that can receive a transmitted message.
We propose a process calculus to study the observational theory of Mobile Ad Hoc Networks. Our calculus is called *Calculus of Mobile Ad Hoc Networks* (CMN). In CMN, a network is modelled as a collection of nodes (which represent devices), running in parallel, and using channels to broadcast messages. Channels can be either public or private to a set of nodes. We write $n[P]_{l,r}^{\mu}$ to denote a node with network address $n$, located at the physical location $l$, with transmission radius $r$, mobility tag $\mu$, and executing the sequential process $P$ which models the behaviour of the device. The location $l$ and the transmission radius $r$ define the cell over which a node can broadcast values using channels; by no means a node is capable to derive its current physical location $l$ or its transmission radius $r$. The mobility tag $\mu$ serves to distinguish between mobile nodes and stationary nodes.

We assume the presence of appropriate protocols to avoid transmission collisions.

The operational semantics of our calculus is given both in terms of a *Reduction Semantics* and in terms of a *Labelled Transition Semantics*, in the SOS style of Plotkin [19]. We prove that the two semantics coincide. Our Labelled Transition System (LTS) captures all the possible interactions of a term with its environment without using any auxiliary discard relation. We then define an appropriate notion of *simulation* and hence of *bisimulation* for MANETs. The concepts of simulation and bisimulation are widely used in the literature for verification purposes: they represent the basis of many verification tools.

The main goal of the paper is to establish when two networks have the same observable behaviour, that is, they are indistinguishable in any context. In this paper, we focus on *reduction barbed congruence* [9], a slight variant of Milner and Sangiorgi’s barbed congruence [14], a branching-time congruence that preserves the observables of the language. The definition of reduction barbed congruence is simple and intuitive. In practise, however, it is difficult to use: the quantification on all contexts is a heavy proof obligation. Simpler proof techniques are based on *labelled bisimilarities* [17, 12]. As a main result, we prove that our (weak) labelled bisimilarity completely characterises reduction barbed congruence. We then use our observational theory to prove a number of non-trivial properties of MANETs.

2 The Calculus

In Table 1, we define the syntax of CMN in a two-level structure, a lower one for *processes* and an upper one for *networks*.

We use letters $m$ and $n$ for *nodes/devices*; $c$ and $d$ for *channels*; $k$ and $l$ for
### Table 1 The Syntax

<table>
<thead>
<tr>
<th>Names:</th>
<th>(a, b, \ldots, k, l, m, n, \ldots \in \mathbb{N})</th>
</tr>
</thead>
<tbody>
<tr>
<td>Networks:</td>
<td>(M, N \ ::= \ 0) empty network</td>
</tr>
<tr>
<td>&amp;</td>
<td>(M_1 \mid M_2) parallel composition</td>
</tr>
<tr>
<td>&amp;</td>
<td>((\nu c)M) channel restriction</td>
</tr>
<tr>
<td>&amp;</td>
<td>(n[P]_{l,r}) node (or device)</td>
</tr>
<tr>
<td>Processes:</td>
<td>(P, Q, R \ ::= \ 0) inactive process</td>
</tr>
<tr>
<td>&amp;</td>
<td>(c(x).P) input</td>
</tr>
<tr>
<td>&amp;</td>
<td>(\overline{c(w)}.P) output</td>
</tr>
<tr>
<td>&amp;</td>
<td>([w_1 = w_2]P, Q) matching</td>
</tr>
<tr>
<td>&amp;</td>
<td>(A(\tilde{v})) recursion</td>
</tr>
<tr>
<td>Mobility tags:</td>
<td>(\mu \ ::= \ m) mobile</td>
</tr>
<tr>
<td>&amp;</td>
<td>(s) stationary</td>
</tr>
</tbody>
</table>

(physical) locations; \(x, y, z\) for variables. Closed values contain the previous entities except for channels and variables. Values include also variables. We use \(u\) and \(v\) for closed values and \(w\) for (open) values. We write \(\tilde{a}\) to denote a tuple \(a_1, \ldots, a_k\) of names.

Networks are collections of nodes (which represent devices), running in parallel, using channels to broadcast messages. Network \(0\) denotes the empty network. \(M_1 \mid M_2\) represents the parallel composition of two networks. In \((\nu c)M\) the channel \(c\) is private to the network \(M\). Unlike other name-passing calculi, such as the \(\pi\)-calculus [13], the restriction operator \((\nu c)M\) models only channel restriction but not channel creation. This is because the number of available channels in a wireless system is standardised by frequency throughout the world (13 for Europe, 11 for North America, and 14 for Japan).

Processes are sequential and live within the nodes. Process \(0\) denotes the inactive processes. The input process \(c(x).P\) can receive any (closed) value \(v\) via channel \(c\) and continue as \(P\), with \(v\) substituted for \(x\). The output process \(\overline{c(v)}.P\) can send the (closed) value \(v\) via channel \(c\) and continue as \(P\). Process \(\[v_1 = v_2\]P, Q\) is the standard “if then else”: it behaves as \(P\) if \(v_1 = v_2\), and as \(Q\) otherwise. We write \(A(\tilde{v})\) to denote a process defined via a (possibly recursive) definition \(A(\tilde{x}) \overset{\text{def}}{=} P\), with \(|\tilde{x}| = |\tilde{v}|\), where \(\tilde{x}\) contains
all channels and variables that appear free in $P$.

Each node has a location and a transmission radius. Nodes cannot be created or destroyed. We write $n[P]_{ll}$ for a node named $n$, located at $l$, with transmission radius $r$, mobility tag $\mu$, and executing process $P$. The node identifier $n$ represents a logical location—the device network address. By contrast, $l$ represents a physical location and, together with the radius $r$, is employed for deriving information about the network connectivity. The mobility tag $\mu$ can be $\mathfrak{m}$ for mobile nodes, and $\mathfrak{s}$ for stationary nodes, i.e. nodes that never change their physical location.

We do not indicate how locations should be specified; for instance, they could be given by means of a coordinate system. In the definition of the operational semantics, we assume the possibility of comparing locations so to determine whether a node lies or not within the transmission cell of another node. We do so by means of a function $d(\cdot, \cdot)$ which takes two locations and returns their distance. In Section 6, we also assume some intuitive meta-operators on locations.

In the process $\tau(w).P$ value $w$ appears in output position; the function $\text{op}(\cdot)$ returns the set of values appearing in output position in a process. In the process $c(x).P$ variable $x$ is bound in $P$, giving rise to the standard notions of $\alpha$-conversion and free and bound variables, denoted with $\text{fv}(\cdot)$ and $\text{bv}(\cdot)$, respectively. Similarly, in a network of the form $(\nu c)M$ the channel name $c$ is bound in $M$ and the notions of $\alpha$-conversion and free and bound channels, $\text{fc}(\cdot)$ and $\text{bc}(\cdot)$, are defined accordingly. We write $\{v/x\}P$ for the capture avoiding substitution of $x$ for $v$ in $P$. We will identify processes and networks up to $\alpha$-conversion. More formally, we will view terms as representatives of their equivalence class with respect to $\equiv_\alpha$, and these representatives will always be chosen so that bound names are distinct from free names.

A (monadic) context $C[\cdot]$ is a network term with a hole, denoted by $[\cdot]$. Contexts are generated by the following grammar:

$$
C[\cdot] ::= \cdot \mid M \mid M \mid (\nu c)[\cdot].
$$

We use a number of notational conventions. Parallel composition of networks has lower precedence with respect to restriction. $\prod_{i \in I} M_i$ means the parallel composition of all networks $M_i$, for $i \in I$. We write $(\nu c)M$ as an abbreviation for $(\nu c_1)\ldots(\nu c_k)M$. We write $\tau(w)$ for $\tau(w).\mathfrak{0}$, and $\mathfrak{0}$ for $n[\mathfrak{0}]_{ll}$. Finally, we write $[w_1 = w_2]P$ for $[w_1 = w_2]P, \mathfrak{0}$.

We assume that there are no free variables in a network (in contrast, there can be free channels). The absence of free variables is trivially maintained as the network evolves. Moreover, as node identifiers denote device network addresses we assume that in any network each node identifier is unique.
2.1 Reduction Semantics

The dynamics of the calculus is specified by the reduction relation over networks, \( \rightarrow \), described in Table 3. As usual in process calculi, the reduction semantics relies on an auxiliary relation, called structural congruence, \( \equiv \), defined in Table 2. Basically, structural congruence brings the participants of a potential interaction into contiguous positions.

Rule (R-Bcast) models the broadcast of a message \( v \) using a channel \( c \). Communication is one-to-many and transmission proceeds even if there is no other process listening for a message: transmission is a non-blocking action. Moreover, as with most process calculi, this communication is deemed to occur instantaneously. Note that when a transmission occurs, some receivers within the range of the transmitter might not receive the message. This may
be due to several reasons such as the presence of obstacles or the asynchrony of nodes. In particular, when $I = \emptyset$ the rule models message loss. In terms of observation this corresponds to a local activity on the network which an observer is not party to. Movement is assumed to be an atomic action: while moving a node cannot do anything else. Rule (R-Move) models arbitrary and unpredictable movements of mobile nodes; notice that stationary nodes cannot move. The remaining rules are standard in process calculi.

The symbol $\rightarrow^*$ denotes the reflexive and transitive closure of $\rightarrow$.

2.2 Behavioural Semantics

In operational semantics two terms are deemed equivalent if they have the same observable behaviour in all possible contexts. So, the question is: What are the “right” observables in our calculus? As in CCS [12] and in $\pi$-calculus [13], we have both transmission and reception of messages. However, unlike those calculi, only the transmission of messages (over unrestricted channels) can be observed. In fact, in a broadcasting calculus an observer cannot see whether a given process actually receive a particular broadcasted value. In particular, if the node $n[c(\langle v \rangle).P]_{l,r}^\mu$ evolves into $n[P]_{l,r}^\mu$, we cannot be sure that some recipient received message $v$ at channel $c$. On the other hand, if a node $n[c(x).P]_{l,r}^\mu$ evolves into $n[\{v/x\}P]_{l,r}^\mu$, then $n$ can be sure that some node has transmitted message $v$ on channel $c$: the network never invents messages!

So, in our calculus the notion of observability is represented by the transmission of messages that can be detected by a pervasive observer i.e. an observer that can listen anywhere, at any channel.

**Definition 2.1 (Barb)** We write $M \downarrow_{c@k}$ if $M \equiv (\nu \tilde{d})(n[\overline{c}(v).P]_{l,r}^\mu | M')$, with $c \not\in \tilde{d}$ and $d(l,k) \leq r$. We write $M \downarrow_{c@k}$ if $M \rightarrow^* M' \downarrow_{c@k}$.

In the following, we use $R$ to denote an arbitrary binary relation over networks. We write $R^= \rightarrow$ to denote the symmetric closure of $R$.

**Definition 2.2** A relation $R$ is **barb preserving** if $M \rightarrow N$ and $M \downarrow_{c@k}$ implies $N \downarrow_{c@k}$.

**Definition 2.3** A relation $R$ is **reduction closed** if $M \rightarrow N$ and $M \rightarrow^* M'$ imply the existence of some $N'$ such that $N \rightarrow^* N'$ and $M' \rightarrow N'$.

**Definition 2.4** A relation $R$ is **contextual** if $M \rightarrow N$ implies $C[M] R C[N]$ for all contexts $C[-]$. 
Table 4 Labelled Transition System - Processes

\[ \begin{align*}
\text{(Input)} & \quad \frac{c(x).P \xrightarrow{cv} \{v/x\}P}{\text{(Then)}} & \quad \frac{P \xrightarrow{\eta} P'}{v = v}\{P, Q \xrightarrow{\eta} P' \}
\text{(Rec)} & \quad \frac{\{\tilde{v}/\tilde{x}\}P \xrightarrow{\eta} P'}{A(\tilde{x}) \xrightarrow{\eta} P'}
\text{(Output)} & \quad \frac{\overline{c}(v).P \xrightarrow{\overline{cv}} P}{\text{(Else)}} & \quad \frac{Q \xrightarrow{\eta} Q'}{v_1 \neq v_2}[v_1 = v_2]P, Q \xrightarrow{\eta} Q'
\end{align*} \]

Finally, everything is in place to define reduction barbed congruence.

**Definition 2.5 (Reduction barbed congruence)** Reduction barbed congruence, written \( \congruence \), is the largest symmetric relation over networks, which is reduction closed, barb preserving, and contextual.

### 3 A Labelled Transition Semantics

Reflecting the language syntax, the Labelled Transition System has two sets of rules: one for processes and one for networks.

Table 4 presents the LTS for processes. Transitions are of the form \( P \xrightarrow{\eta} P' \), where \( \eta \) ranges over input and output actions. More precisely, \( cv \) and \( \overline{cv} \) denote, respectively, input and output of a closed value \( v \) at channel \( c \). The rules in Table 4 are self-explanatory.

Table 5 contains the LTS for networks. Transitions are of the form \( M \xrightarrow{\lambda} M' \), where the grammar for \( \lambda \) is:

\[ \lambda \ ::= \ c?v@l \mid c!v[l, r] \mid c!v@K \mid \tau \]

Rule (Rcv) models the reception at \( l \) of message \( v \) via channel \( c \). Rule (Snd) models the broadcast, with transmission radius \( r \), of message \( v \) via channel \( c \), from a node located at \( l \). Rule (Bcast) models the propagation of broadcast. The requirement \( d(l, l') \leq r \) guarantees that only nodes within the transmission cell of the transmitter may hear the communication. Rule (Obs) models the fact that every action \( c!v[l, r] \) can be detected (and hence observed) by any node located in the transmission cell at \( l \) with radius \( r \).

The action \( c!v@K \) represents the transmission of message \( v \) via channel \( c \) to
Table 5  Labelled Transition System - Networks

<table>
<thead>
<tr>
<th>Rule</th>
<th>Transition</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Rcv)</td>
<td>$P \xrightarrow{n[P]_l^\mu} P'$</td>
</tr>
<tr>
<td>(Snd)</td>
<td>$P \xrightarrow{n[P]_l^\mu} P'$</td>
</tr>
<tr>
<td>(Bcast)</td>
<td>$M \xrightarrow{c[l,r]} M' \quad N \xrightarrow{c[l,r]} N'$</td>
</tr>
<tr>
<td>(Obs)</td>
<td>$M \xrightarrow{c[l,r]} M' \quad K \subseteq {k : d(l,k) \leq r} \quad K \neq \emptyset$</td>
</tr>
<tr>
<td>(Par)</td>
<td>$M \xrightarrow{\lambda} M'$</td>
</tr>
<tr>
<td>(Move)</td>
<td>$n[P]<em>{l,r} \xrightarrow{\tau} n[P]</em>{l,r}$</td>
</tr>
<tr>
<td>(Res)</td>
<td>$M \xrightarrow{\lambda} M' \quad c \notin fc(\lambda)$</td>
</tr>
</tbody>
</table>

1. If $M \xrightarrow{c[l,r]} M'$ then there are $n$, $P$, $\mu$, $l$, $r$, $M_1$, and $\tilde{d}$, with $c \notin \tilde{d}$, such that $M \equiv (\nu\tilde{d})(n[c(x).P]_{l,r} | M_1)$ and $M' \equiv (\nu\tilde{d})(n[\tilde{v}[x].P]_{l,r} |$
2. If $M \xrightarrow{c[l,r]} M'$ then there are $n, P, \mu, l, r, M_1, I$ (possibly empty) and $\bar{d}$, with $c \notin \bar{d}$, and $n_i, P_i, \mu_i, r_i, l_i$, with $d(l_i, l_i) \leq r$, for all $i \in I$, such that

$$M \equiv (\nu \bar{d})(n[\bar{c(v)}.P]_{l,r}^{\mu} \mid \prod_{i \in I} n_i[c(x_i).P_i]_{l_i,r_i}^{\mu_i} \mid M_1)$$

and

$$M' \equiv (\nu \bar{d})(n[P]_{l,r}^{\mu} \mid \prod_{i \in I} n_i[\{v/x_i\}P_i]_{l_i,r_i}^{\mu_i} \mid M_1).$$

**Proof** By induction on the transitions rules of Table 5.

We also need to show that structural congruence respects the transitions of Table 5.

**Lemma 3.2 (≡ respects transitions)** If $M \xrightarrow{\lambda} M'$ and $M \equiv N$ then there exists $N'$ such that $N \xrightarrow{\lambda} N'$ and $M' \equiv N'$.

**Proof** We outline the proof, which proceeds by induction on the depth of the inference $M \xrightarrow{\lambda} M'$. It is clearly enough to prove the result in the special case that $M \equiv N$ is due to a single application of a structural rule from Table 2; the general case follows just by iterating the special case.

The full proof must treat all possible cases for the final step of the inference $M \xrightarrow{\lambda} M'$. Here we consider just one case; suppose that it is inferred by rule (Par), where $M$ is $M_1 \mid M_2$ and $M'$ is $M_1' \mid M_2$. with $M_1 \xrightarrow{\lambda} M_1'$ inferred by a shorter inference. Now there are many ways in which $M_1 \mid M_2 \equiv Q$ may be due to a single use of a structural congruence rule; we will confine ourselves to considering just two cases.

**Case 1.** Suppose that the commutativity rule of Table 2 is used, so that $N$ is $M_2 \mid M_1$. In this case we use the rule (Par) to deduce that $N \xrightarrow{\lambda} M_2 \mid M_1'$. Now, take $N'$ to be $M_2 \mid M_1'$; we have $M' \equiv N'$, as required.

**Case 2.** Suppose that a single rule of structural congruence is used within $M_1$, so that $M_1 \equiv N_1$ and $N$ is $N_1 \mid M_2$. Then, since $M_1 \xrightarrow{\lambda} M_1'$ is inferred by a shorter inference, by appeal to induction we have $N_1 \xrightarrow{\lambda} N_1'$ and $M_1' \equiv N_1'$. Now take $N'$ to be $N_1' \mid M_2$; by using rule (Par) we deduce that $N \xrightarrow{\lambda} N'$ and $M' \equiv N'$, as required.

So the result follows by a fairly lengthy case analysis, both for the structural congruence rule used and for the last step of the transition inference. □
Theorem 3.3 (Harmony Theorem)

1. If $M \xrightarrow{\tau} M'$ then $M \rightarrow M'$.

2. If $M \rightarrow M'$ then $M \xrightarrow{\tau} \equiv M'$.

Proof. The first part is by induction on the derivation $M \xrightarrow{\tau} M'$. We recall that the $\tau$-transitions can only be generated by the rules in Table 5.

Suppose that the $\tau$-action has been generated by an application of rule (Lose). In this case, we have $M \xrightarrow{[\mathrm{clv}[l,r]]} N$ for some $c$, $v$, $l$, and $r$. By an application of Lemma 3.1 we get:

$$M \equiv (\nu \tilde{d})(n[\tilde{c}(v).P]^\mu_{l,r} | \prod_{i \in I} n_i[c(x_i).P_i]^\mu_{l_i,r_i} | M_1)$$

and

$$M' \equiv (\nu \tilde{d})(n[P]^\mu_{l,r} | \prod_{i \in I} n_i[\{v/x_i\}P_i]^\mu_{l_i,r_i} | M_1)$$

for some $n$, $v$, $P$, $\mu$, $l$, $r$, $M_1$, $\tilde{d}$, with $c \notin \tilde{d}$, and some $n_i$, $P_i$, $\mu_i$, $r_i$, $l_i$, such that $d(l, l_i) \leq r_i$ for all $i \in I$. By applying rules (R-Bcast), (R-Par), and (R-Res) we get

$$(\nu \tilde{d})(n[\tilde{c}(v).P]^\mu_{l,r} | \prod_{i \in I} n_i[c(x_i).P_i]^\mu_{l_i,r_i} | M_1) \xrightarrow{} (\nu \tilde{d})(n[P]^\mu_{l,r} | \prod_{i \in I} n_i[\{v/x_i\}P_i]^\mu_{l_i,r_i} | M_1) .$$

By applying rule (R-Struct) we obtain $M \rightarrow M'$, as required.

Suppose now that that the $\tau$-action $M \xrightarrow{\tau} M'$ has been generated by an application of rule (Move). Then, by an application of rule (R-Move) we derive that $M \rightarrow M'$, as required.

The other cases follow straightforwardly from the congruence rules of the reduction relation.

The second part of the statement is proved by induction of the derivation $M \rightarrow M''$.

Suppose that the derivation $M \rightarrow M'$ has been generated by an application of rule (R-Bcast), that is,

$$n[\tilde{c}(v).P]^\mu_{l,r} | \prod_{i \in I} n_i[c(x_i).P_i]^\mu_{l_i,r_i} \rightarrow n[P]^\mu_{l,r} | \prod_{i \in I} n_i[\{v/x_i\}P_i]^\mu_{l_i,r_i}$$
such that $d(l_i, l_i) \leq r$, for all $i \in I$. Then, the derivation below is valid.

$$\tau(v).P \xrightarrow{cv} P \quad c(x_1).P_1 \xrightarrow{cr} \{v/x_1\}P_1$$

$$n[\tau(v).P]_{l,r}^{\mu} \xrightarrow{c[l][r]} n[P]_{l,r}^{\mu} \quad n_1[c(x_1).P_1]_{l_1,r_1}^{\mu_1} \xrightarrow{c[l][r]} n_1[\{v/x_1\}P_1]_{l_1,r_1}^{\mu_1}$$

$$n[\tau(v).P]_{l,r}^{\mu} | n_1[c(x_1).P_1]_{l_1,r_1}^{\mu_1} \xrightarrow{c[l][r]} n[P]_{l,r}^{\mu} | n_1[\{v/x_1\}P_1]_{l_1,r_1}^{\mu_1}$$

By applying $| I | - 1$ times rule (Beast) and one time rule (Lose) we get:

$$n[\tau(v).P]_{l,r}^{\mu} | \prod_{i \in I} n_i[c(x_i).P_i]_{l_i,r_i}^{\mu_i} \xrightarrow{\tau} n[P]_{l,r}^{\mu} | \prod_{i \in I} n_i[\{v/x_i\}P_i]_{l_i,r_i}^{\mu_i}$$

as required.

Suppose that the derivation $M \xrightarrow{\tau} M'$ has been generated by an application of rule (R-Move), in this case, by an application of rule (Move) we have $M \xrightarrow{\tau} M'$.

Suppose that the derivation $M \xrightarrow{\tau} M'$ has been generated by an application of rule (R-Struct) $M \equiv N \xrightarrow{\tau} N' \equiv M'$.

The induction hypothesis tells us that there is $N''$ such that $N \xrightarrow{\tau} N'' \equiv N'$. Lemma 3.2 tells us that there is $M''$ such that $M \xrightarrow{\tau} M''$ and $M'' \equiv N''$. By transitivity of $\equiv$, it follows that $M \xrightarrow{\tau} M'$, as required.

Finally, as both the $\tau$-transitions and the structural congruence are preserved by networks contexts, the cases when the reduction $M \xrightarrow{\tau} M'$ is derived either by rule (R-Par) or by rule (R-Res) are straightforward. □

4 Bi-simulation Proof Methods

In this section, we use our LTS define an appropriate notion of simulation/bisimulation for ad hoc networks. We then prove that our labelled bisimilarity implies reduction barbed congruence, and hence represents a valid method for proving that two networks are reduction barbed congruent.

For commodity, we use the metavariable $\alpha$ to range over those actions that will be used in the definition of (bi)simulation. Formally,

$$\alpha ::= c?v@l \mid c?v@K \mid \tau .$$

Since we are interested in weak behavioural equivalences, that abstract over $\tau$-actions, we introduce the notion of weak action. The definition is not completely standard:
\begin{itemize}
  \item \( \Rightarrow \) denotes the reflexive and transitive closure of \( \tau \rightarrow \);
  \item \( c?v@l \) denotes \( \Rightarrow c?v@l \Rightarrow \);
  \item \( c?v@K \) denotes \( \Rightarrow c?v@K_1 \Rightarrow \ldots \Rightarrow c?v@K_n \Rightarrow \), for \( \bigcup_{i=1}^{n} K_i = K \);
  \item \( \overset{\hat{\alpha}}{\Rightarrow} \) denotes \( \text{if } \alpha = \tau \text{ and } \overset{\alpha}{\Rightarrow} \text{otherwise.} \)
\end{itemize}

Notice that the definition of the weak observable action \( c?v@l \) may contain several (strong) observable actions of the form \( c?v@K_i \). This is because a distributed observer that receives in several computation steps an instance of message \( v \) at each location in \( K \) cannot assume that those messages belong to the same multicast send.

**Definition 4.1 (Bisimilarity)** A binary relation \( \mathcal{R} \) over networks is a simulation if \( M \mathcal{R} N \) implies:

\begin{itemize}
  \item If \( M \overset{\alpha}{\rightarrow} M' \), \( \alpha \neq c?v@l \), then there is \( N' \) such that \( N \overset{\hat{\alpha}}{\Rightarrow} N' \) and \( M' \mathcal{R} N' \);
  \item If \( M \overset{c?v@l}{\rightarrow} M' \) then there is \( N' \) such that:
    \begin{itemize}
      \item either \( N \overset{c?v@l}{\Rightarrow} N' \) and \( M' \mathcal{R} N' \)
      \item or \( \Rightarrow N' \) and \( M' \mathcal{R} N' \).
    \end{itemize}
\end{itemize}

We say that \( N \) simulates \( M \) if there is some simulation \( \mathcal{R} \) such that \( M \mathcal{R} N \). A relation \( \mathcal{R} \) is called bisimulation if both \( \mathcal{R} \) and its converse are simulations.

We say that \( M \) and \( N \) are bisimilar, written \( M \approx N \), if there is some bisimulation \( \mathcal{R} \) such that \( M \mathcal{R} N \).

Notice that, since reception of messages cannot be directly detected, the clause for message reception imposes weaker requirements, allowing to match input actions with \( \tau \)-actions.

It is easy to show that our labelled bisimilarity is an equivalence relation. However, our bisimilarity enjoys a much more important property: the closure under contexts.

**Lemma 4.2 (\( \approx \) is contextual)** Let \( M \) and \( N \) be two networks such that \( M \approx N \). Then,

1. \( M | O \approx N | O \), for all networks \( O \);
2. \( (\nu c)M \approx (\nu c)N \), for all channels \( c \).
Proof. As regards the first item, i.e that $\approx$ is preserved by parallel composition, we prove that the relation

$$\mathcal{S} \overset{\text{def}}{=} \{(M \mid O, N \mid O) \mid \text{for all } O \text{ such that } M \approx N\}$$

is a bisimulation. We do a case analysis on the transition $M \mid O \xrightarrow{\alpha} \hat{M}$. The interesting cases are when the transition is due to an interaction between $M$ and $O$, i.e. when rule (Bcast) is used.

Let $M \mid O \xrightarrow{\text{cle}[l,r]} \hat{M}$ because $M \mid O \xrightarrow{\text{cle}[l,r]} \hat{M}$ for some $l$ and $r$, with $d(l, k) \leq r$, for all $k \in K$ due to an application of rule (Bcast). There are two possibilities:

- $M \mid O \xrightarrow{\text{cle}[l,r]} \hat{M}$ because $M \xrightarrow{\text{cle}[l,r]} M'$ and $O \xrightarrow{\text{cle}[l,r]} O'$, with $d(l, l') \leq r$ and $\hat{M} = M' \mid O'$. In this case, by an application of rule (Obs) we have $M \xrightarrow{\text{cle}[K]} M'$, with $K' = K \cup \{l'\}$. As $M \approx N$ there is $N'$ such that $N \xrightarrow{\text{cle}[K']} N'$ with $M' \approx N'$. By applying rule (Obs) backward there must be $K_1, \ldots, K_n$ such that $N \xrightarrow{\text{cle}[K_1]} \ldots \xrightarrow{\text{cle}[K_n]} N'$ with $\bigcup_{i=1}^{n} K_i = K'$ and $l' \in K_j$, for some $1 \leq j \leq n$. This implies that

$$N \xrightarrow{\text{cle}[K_1]} \ldots \xrightarrow{\text{cle}[K_n]} \Rightarrow N'$$

with $d(l_j, k) \leq r_j$, for all $k \in K_j$. Hence by an application of rule (Bcast):

$$N \mid O \xrightarrow{\text{cle}[K_1]} \ldots \xrightarrow{\text{cle}[K_n]} \Rightarrow N' \mid O'.$$

Finally, by applying rule (Obs) we can turn the transition $\xrightarrow{\text{cle}[l,r]}$ in $\xrightarrow{\text{cle}[K_j]}$. This implies $N \mid O \xrightarrow{\text{cle}[K]} N' \mid O'$ with $(M' \mid O', N' \mid O') \in \mathcal{S}$, as required.

- $M \mid O \xrightarrow{\text{cle}[l,r]} \hat{M}$ because $M \xrightarrow{\text{cle}[l,r]} M'$ and $O \xrightarrow{\text{cle}[l,r]} O'$, with $d(l, l') \leq r$ and $\hat{M} = M' \mid O'$. As $M \approx N$ there is $N'$ such that:

- either $N \xrightarrow{\text{cle}[l,r]} N'$, with $M' \approx N'$; in this case

$$N \mid O \xrightarrow{\text{cle}[l,r]} \Rightarrow N' \mid O'$$

and, by rule (Obs), also $N \mid O \xrightarrow{\text{cle}[K]} N' \mid O'$, with $(M' \mid O', N' \mid O') \in \mathcal{S}$, as required.
– or \( N \Rightarrow N' \), with \( M' \approx N' \); in this case, by applying rule (Par) we obtain \( N \mid O \Rightarrow \stackrel{c_{v[l,r]}}{\Rightarrow} N' \mid O' \) and, by rule (Obs) also \( N \mid O \Rightarrow \stackrel{c_{v[\alpha]K}}{\Rightarrow} N' \mid O' \), with \( (M' \mid O', N' \mid O') \in S \), as required.

Let \( M \mid O \xrightarrow{\tau} \hat{M} \) because \( M \mid O \Rightarrow \stackrel{c_{v[l,r]}}{\Rightarrow} \hat{M} \). We reason as in the previous case.

The remaining cases, when there is no interaction between \( M \) and \( O \), are easy to deal with.

In order to prove that \( \approx \) is preserved by restriction, it suffices to show that the relation

\[
S \overset{\text{def}}{=} \{(\nu c)M , (\nu c)N) \text{ for all } c \text{ such that } M \approx N \}
\]

is a bisimulation. We do a case analysis on the transition \((\nu c)M \xrightarrow{\alpha} O\). The proof is straightforward as channels cannot be transmitted and hence there is no scope extrusion.

In the following lemma we point out a close relationship between the observation predicate \( \downarrow_{c\alpha k} \) and a specific action.

**Lemma 4.3**

1. If \( M \xrightarrow{c_{v[l,r]}} M' \) then \( M \downarrow_{c\alpha k} \), for all \( k \in K \);

2. if \( M \downarrow_{c\alpha k} \) then there is a value \( v \) and a set of locations \( K \), with \( k \in K \), such that \( M \xrightarrow{c_{v[\alpha]K}} M' \).

We can now prove that our bisimilarity is a proof methods for reduction barbed congruence, i.e. that \( \approx \) is contained in \( \cong \).

**Theorem 4.4 (Soundness)** Let \( M \) and \( N \) be two arbitrary networks such that \( M \approx N \), then \( M \cong N \).

**Proof** We recall that \( \cong \) is the least symmetric relation which is reduction closed, barb-preserving, and contextual. In fact, the bisimilarity is reduction closed (using Theorem 3.3), barb-preserving (by Lemma 4.3), and contextual (by Lemma 4.2). Thus, \( \approx \subseteq \cong \). \( \square \)
5 Characterising Reduction Barbed Congruence

In this section, we prove that our labelled bisimilarity is more than a proof technique. Actually, it represents a complete characterisation of reduction barbed congruence.

When proving the completeness result, i.e. that reduction barbed congruence is contained in the labelled bisimilarity, we implicitly use a standard property of reduction barbed congruence.

**Proposition 5.1** If \( M \cong N \) then

- \( M \Downarrow_{c:k} \iff N \Downarrow_{c:k} \)
- \( M \Rightarrow M' \implies \text{there is } N' \text{ such that } N \Rightarrow N' \text{ and } M' \cong N' \).

**Lemma 5.2 (Completeness)** Reduction barbed congruence is contained in the bisimilarity.

**Proof** We prove that the relation \( R = \{(M, N) \mid M \cong N\} \) is a bisimulation. The result will then follow by co-induction.

- Suppose that \( M R N \) and \( M \xrightarrow{\tau} M' \). This case is easy to deal with.

- Suppose that \( M R N \) and \( M \xrightarrow{\text{clv@K}} M' \), with \( K = \{k_1, \ldots, k_n\} \). As the action \( \text{clv@K} \) can only be generated by an application of rule (Obs), it follows that \( M \xrightarrow{\text{clv}[l,r]} M' \) for some \( l \) and \( r \) such that \( d(l, k) \leq r \), for all \( k \in K \).

Let us build up a context which mimics the effect of the action \( \text{clv@K} \), and also allows us to subsequently compare the residuals of the two systems under consideration. Our context has the form:

\[
C[] \overset{\text{def}}{=} [\cdot] \prod_{i=1}^{n} (m_i[c(x).[x = v]\overline{f_i}(x)]_{k_i,r_i}^{s} \mid n_i[f_i(x).\overline{ok_i}(x)]_{k_i,r_i}^{s})
\]

with names \( m_i, n_i \), for \( 1 \leq i \leq n \), and channel names \( f_i \) and \( \text{ok}_i \), for \( 1 \leq i \leq n \), fresh. Intuitively, the existence of the barbs on the fresh channels \( f_i \) indicates that the action has not yet happened, whereas the presence of the barbs on channels \( \text{ok}_i \), together with the absence of the barbs on \( f_i \), ensures that the action has been performed.
As \( \cong \) is preserved by network contexts, \( M \cong N \) implies \( C[M] \cong C[N] \).
As \( M \xrightarrow{\text{cl}(l,r)} M' \), it follows that
\[
C[M] \Rightarrow M' \mid \prod_{i=1}^{n} (m_i[0]_{k_i,r_i} \mid n_i[\overline{ok_i}(v)]_{k_i,r_i}) = \hat{M}
\]
with \( \hat{M} \not\Downarrow f_{i,k_i} \) and \( \hat{M} \not\Downarrow ok_i \), for \( 1 \leq i \leq n \).

The reduction sequence above must be matched by a corresponding reduction sequence \( C[N] \Rightarrow \hat{N} \) with \( \hat{M} \cong \hat{N} \), \( \hat{N} \not\Downarrow f_{i,k_i} \) and \( \hat{N} \not\Downarrow ok_i \), for \( 1 \leq i \leq n \).

The constrains on the barbs allow us to deduce the structure of the above reduction sequence. That is:
\[
C[N] \Rightarrow N' \mid \prod_{i=1}^{n} (m_i[0]_{k_i,r_i} \mid n_i[\overline{ok_i}(v)]_{k_i,r_i}) \cong \hat{N} .
\]

This implies that \( N \xrightarrow{\text{cl}(L)} N' \), with \( K \subseteq L \). More precisely, the derivative \( N' \) might be reached performing several outputs of message \( v \) along the same channel \( c \). However, as all nodes \( m_i \) are reached by a transmission along channel \( c \) coming from \( N \), we can be sure that \( K \subseteq L \). It is then easy to show that \( N \xrightarrow{\text{cl}(K)} N' \) by considering in the composition of the weak action only on those outputs addressed to the locations in \( K \), and turning the others in \( \tau \)-actions using rule (Lose).

As \( \hat{M} \cong \hat{N} \) and reduction barbed congruence is preserved by restriction, we have
\[
(\nu \bar{f}, \bar{ok})\hat{M} \cong (\nu \bar{f}, \bar{ok})\hat{N} .
\]

As channels \( f_i \) and \( ok_i \), for \( 1 \leq i \leq n \), are fresh we have
\[
- (\nu \bar{f}, \bar{ok})\hat{M} \equiv M' \mid (\nu \bar{f}, \bar{ok})(\prod_{i=1}^{n} m_i[0]_{k_i,r_i} \mid n_i[\overline{ok_i}(v)]_{k_i,r_i})
- (\nu \bar{f}, \bar{ok})\hat{N} \equiv N' \mid (\nu \bar{f}, \bar{ok})(\prod_{i=1}^{n} m_i[0]_{k_i,r_i} \mid n_i[\overline{ok_i}(v)]_{k_i,r_i}) .
\]

Using our labelled bisimilarity and Theorem 4.4 is easy to prove that
\[
(\nu \bar{f}, \bar{ok})(\prod_{i=1}^{n} m_i[0]_{k_i,r_i} \mid n_i[\overline{ok_i}(v)]_{k_i,r_i}) \cong 0 .
\]

As a consequence, it follows that \( M' \cong N' \), as required.
• Suppose that $M \mathcal{R} N$ and $M \xrightarrow{c?v@l} M'$. We recall that this actions cannot be directly observed, as exemplified by the presence of weaker requirements in the clause for inputs in Definition 4.1. However, a context associated to the action $c?v@l$ could be

$$C[\cdot] \overset{\text{def}}{=} [\cdot] | n[\overline{c}(v).\overline{f}(v).\overline{ok}(v)]_{k,r}^s$$

with $f$ and $ok$ fresh channels, and $d(l,k) \leq r$.

As $\cong$ is preserved by network contexts, $M \cong N$ implies $C[M] \cong C[N]$. As $M \xrightarrow{c?v@l} M'$, it follows that if

$$C[M] \Rightarrow M' | n[\overline{ok}(v)]_{k,r}^s = \hat{M}$$

with $\hat{M} \not\Downarrow f@k$ and $\hat{M} \Downarrow ok@k$.

The reduction sequence above must be matched by a corresponding reduction sequence $C[N] \Rightarrow \hat{N}$ with $\hat{M} \cong \hat{N}$, $\hat{N} \not\Downarrow f@k$ and $\hat{N} \Downarrow ok@k$. The constrains on the barbs allow us to deduce the structure of the above reduction sequence. That is:

$$C[N] \Rightarrow N' | n[\overline{ok}(v)]_{k,r}^s \cong \hat{N}$$

However, this does not ensure us that $N$ actually performed the $c?v@l$ actions. We can only conclude that there is $N'$ such that either $N \xrightarrow{c?v@l} N'$ or $N \Rightarrow N'$, in case rule (Lose) has been applied to node $n$.

As $\hat{M} \cong \hat{N}$ and $\cong$ is preserved by restriction it follows that

$$(\nu ok)\hat{M} \cong (\nu ok)\hat{N}$$

from which we can easily derive $M' \cong N'$, as required.

□

An easy consequence of Theorem 4.4 and Lemma 5.2 is the following.

**Theorem 5.3 (Characterisation)** Bisimilarity and reduction barbed congruence coincide.

### 6 Properties and examples

In this section, we prove a number of properties using our observational theory.

We start proving an interesting feature of mobile nodes.
Theorem 6.1 (Ubiquity of mobile nodes) For any process \( P \), physical locations \( k \) and \( l \), and transmission radius \( r \), it holds that

\[ n[P]_{k,r}^m \approx n[P]_{l,r}^m. \]

Proof We show that the relation

\[ S \overset{\text{def}}{=} \{ (n[P]_{k,r}^m, n[P]_{l,r}^m) : \forall k, l, \forall P \} = \cup I \]

is a bisimulation, where \( I \) is the identity relation. \( \square \)

The next result shows that silent nodes cannot be detected (or observed). A node is said silent if it never transmit messages.

Theorem 6.2 (Silent nodes cannot be observed) If process \( P \) does not contain output constructs, then

\[ n[P]_{l,r}^\mu \approx 0 \]

for any \( l \) and \( r \).

Proof It follows from our definition of bisimilarity in which it is possible to match both \( \tau \)-actions and input actions with weak \( \tau \)-actions. We recall that \( \Rightarrow \) is the reflexive and transitive closure of \( \rightarrow \). \( \square \)

Now, we show how syntactically different infinite output sequences may be semantically indistinguishable, because of message loss.

Theorem 6.3 (Mixing up infinite output sequences) Let \( \text{ALT}(a, b) \overset{\text{def}}{=} \overline{a}(\overline{a}.\overline{b}).\text{ALT}(a, b) \). Then, for any \( l, n, r, u, \) and \( v \) it holds that:

1. \( n[\text{ALT}(u, v)]_{l,r}^u \approx n[\text{ALT}(v, u)]_{l,r}^u \)
2. \( n[\text{ALT}(u, v)]_{k,r}^u \approx n[\text{ALT}(v, u)]_{l,r}^u \).

Proof We only prove the second statement, the first one is simpler. We show that the relation

\[ R \overset{\text{def}}{=} \{ (n[\text{ALT}(u, v)]_{k,r}^u, n[\text{ALT}(v, u)]_{l,r}^u) : \text{ for all } k, l \} = \cup I \]

where \( I \) is the identity relation, is a bisimulation up to \( \equiv \). \( \square \)
This result can be generalised replacing \( u \) and \( v \) with an arbitrary finite set \( V = \{v_1, \ldots, v_n\} \) of messages. More generally, if two nodes contain only an infinite sequence of output constructs transmitting values belonging to some finite set \( V \), such that for each \( v \in V \) the output \( \overline{v}(v) \) appears an infinite number of times, then the two nodes are equivalent.

In the next result, we show that devices transmitting messages “ad infinitum” may obfuscate the transmission activity of nodes which are transmitting the same messages within the same transmission cell. We recall that the function \( \text{fc}(\cdot) \) returns the set of free channels contained in one or more processes, while \( \text{op}(\cdot) \) returns the set of values appearing in output position in one or more processes.

**Theorem 6.4 (Obfuscating message transmission)** Let \( P \) and \( Q \) be two processes such that \( \text{fc}(P, Q) \subseteq \{c\} \), for some channel \( c \), and \( \text{op}(P, Q) \subseteq \{u, v\} \), for some values \( u \) and \( v \). Let \( \text{ALT}(a, b) \overset{\text{def}}{=} \overline{\overline{a}}(a).\overline{\overline{b}}(b).\overline{\text{ALT}}(a, b) \).

Then,

1. \( n[P]^a_{l,r} | m[\text{ALT}(u, v)]^a_{l,r} \approx n[Q]^a_{l,r} | m[\text{ALT}(u, v)]^a_{l,r} \)
2. \( n[P]^a_{k,r} | m[\text{ALT}(u, v)]^a_{k,r} \approx n[Q]^{a'}_{k',r} | m[\text{ALT}(u, v)]^{a'}_{k',r} \).

**Proof**

Let us prove the first part. By transitivity of \( \approx \), it suffices to prove that

\[
\forall P. \text{fc}(P) \subseteq \{c\} \land \text{op}(P) \subseteq \{u, v\}.
\]

for all \( l \) and \( r \), and for all \( P \) such that \( \text{fc}(P) \subseteq \{c\} \) and \( \text{op}(P) \subseteq \{u, v\} \). At this purpose, we show that the binary relation

\[
\{(n[P]^a_{l,r} | m[\text{ALT}(u, v)]^a_{l,r}, m[\text{ALT}(u, v)]^a_{l,r}) : \forall P. \text{fc}(P) \subseteq \{c\} \land \text{op}(P) \subseteq \{u, v\}\} =
\cup
\{(n[P]^a_{l,r} | m[\text{ALT}(u, v)]^a_{l,r}, m[\text{ALT}(u, v)]^a_{l,r} : \forall P. \text{fc}(P) \subseteq \{c\} \land \text{op}(P) \subseteq \{u, v\}\} =
\]

is a bisimulation.

The proof of the second part is similar to the previous case, except that the bisimulation is parameterised on the possible locations \( k, l, k', \) and \( l' \).

Again, the previous theorem can be generalised to deal with a finite set of values.

The next results are about range amplifiers and range repeaters, and make particularly sense for stationary nodes, like access points.

Suppose we want to amplify the signal coming from a node \( n[P]^a_{l,r} \). In particular, suppose we are interested in having a transmission cell with a radius \( r' \), where \( r' > r \). In this case, we could place our node side by side with
an amplifier node that simply repeats the signal with transmission radius $r'$. In such a scenario, if node $n$ is single-channel, i.e., it uses only one channel, then the whole system simulates the node $n$ equipped with a more powerful antenna, with transmission radius $r'$, i.e., $n[P]_{l,r}$.

We write $ra[c \leftrightarrow d]_{l,r}$ to denote a range amplifier working on channel $c$, where the process $c \leftrightarrow c$ is a forwarder process whose general recursive definition is

$$a \leftrightarrow b \overset{\text{def}}{=} a(x).b(x).a \leftrightarrow b$$

This process receives values at channel $a$ and retransmits them on channel $b$; in $c \leftrightarrow c$ the same channel $c$ is used for reception and transmission. We will use the definition of forwarder process in several examples.

**Theorem 6.5 (Range amplifiers)** Let $P$ be a process such that $\text{fc}(P) \subseteq \{c\}$, for some channel $c$. Let $l$ be an arbitrary physical location, and $r$ an arbitrary transmission radius. Then, for any transmission radius $r' > r$, the system

$$n[P]_{l,r} \mid ra[c \leftrightarrow d]_{l,r}$$

simulates the node $n[P]_{l,r'}$.

**Proof** We prove that the relation

$$\mathcal{S} \overset{\text{def}}{=} \{(n[P]_{l,r'} \mid n[P]_{l,r} \mid ra[c \leftrightarrow d]_{l,r'}): \forall l \forall P. \text{fc}(P) \subseteq \{c\}\}$$

is a simulation. We do a case analysis on the structure of $P$. The only interesting case is when $P = \varepsilon(v).P'$.

Nodes like $ra[c \leftrightarrow d]_{l,r}$ actually model a well-know device in the literature, called range repeater (or range extender). In general, a repeater simply regenerates a network signal in order to extend the range of the existing network infrastructure. In a wireless networks a range repeater does not physically connect by wire to any part of the network. Instead, it receives radio signals from an access point, end user device, or another repeater and retransmits the frames. This makes it possible for a repeater located in between an access point and a distant stationary user to act as a relay for frames travelling back and forth between the user and the access point. In this manner, using a range repeater, a distant stationary user can get connected to the network. For instance, suppose we want to extend the range of an access point $n[P]_{k,r}$. In particular, suppose we want to cover the cell located at $l$ with radius $r'$. In this case, if $d(k, l) \leq r$ and $d(k, l) \leq r'$ we could add a range repeater at $l$ that simply repeats the signal back and forth with transmission radius $r'$. In such a scenario, if node $n$ is single-channel, i.e., it uses only one channel, then the introduction of the range repeater allows us
to simulate the presence of the access point \( n \) at \( l \) with transmission radius \( r' \), i.e. \( n[P]_{l,r'}^s \).

**Theorem 6.6 (Range repeaters)** Let \( P \) be a process such that \( \text{fc}(P) \subseteq \{c\} \), for some channel \( c \). Let \( k, l \) be physical locations, and \( r, r' \) be transmission radii such that \( d(k, l) \leq r \) and \( d(k, l) \leq r' \). Then, the system

\[
\text{simulates the node } n[P]_{l,r'}^s.
\]

**Proof** We prove that the relation

\[
\{(n[P]_{l,r'}^s, n[P]_{k,r}^s | \text{rr}[c \leftrightarrow c]_{l,r'}^s) : \forall k, l. d(k, l) \leq r \land d(k, l) \leq r', \forall P. \text{fc}(P) \subseteq \{c\}\}
\]

is a simulation. \( \square \)

A well-known downside of range repeaters, though, is that they reduce the throughput of the network. A range repeater must receive and retransmit each frame on the same radio frequency channel, which effectively doubles the number of frames that are sent. In particular, accordingly with the protocol CSMA/CA, whenever the range repeater transmits on channel \( c \) the node \( n \) must remain silent to avoid collisions. A way to avoid this inconvenient could be that of using more sophisticated range repeaters working on two different channels: for example, channel \( c \) for communicating with the access point \( n \), and a different channel, say \( d \), to interact with the local stationary users.

**Theorem 6.7 (Range repeaters with two channels)** Let \( P \) be a process such that \( \text{fc}(P) \subseteq \{c\} \), for some channel \( c \). Let \( k, l \) be physical locations, and \( r, s \) be transmission radii, such that \( d(k, l) \leq r \) and \( d(k, l) \leq r' \). Then, for any channel \( d \), the system

\[
\text{simulates the node } n[\{d/c\}P]_{l,r'}^s.
\]

**Proof** We prove that the relation

\[
\{n[\{d/c\}P]_{l,r'}^s, n[P]_{k,r}^s | \text{out}[c \leftrightarrow d]_{l,r'}^s | \text{in}[d \leftrightarrow c]_{l,r'}^s) : \forall k, l. d(k, l) \leq r \land d(k, l) \leq r', \forall P. \text{fc}(P) \subseteq \{c\}\}
\]

is a simulation. \( \square \)
As already pointed out, the previous results on range amplifiers and range repeaters make particular sense when dealing with stationary nodes. In fact, when dealing with mobile nodes those devices are basically superfluous, as exemplified in the following statement.

**Theorem 6.8** Let \( k, l \) be physical locations and \( r, r' \) be transmission radii such that \( r \geq r' \). Then,

\[
n[P]_{k,r}^p \simulates n[P]_{l,r'}^p .
\]

**Proof** We show that the relation

\[
S \overset{\text{def}}{=} \{(n[P]_{l,r'}^p, n[P]_{k,r}^p) : \forall k, l \forall P \}
\]

is a simulation. \( \square \)

Finally, we provide a result concerning with energy consumption. It is well-know \[24\] that the power \( p_k \) required by a node located at \( k \) to correctly transmit data to a node located at \( l \) must satisfy the inequality \( \frac{p_k}{d(k,l)^\alpha} \geq \beta \), where \( \alpha \geq 2 \) is the *distance-power gradient* and \( \beta \geq 1 \) is the *transmission quality* parameter.\(^1\) While the value of \( \beta \) is usually set to 1, the value of \( \alpha \) depends on environmental conditions. In the ideal case, we have \( \alpha = 2 \); however \( \alpha \) is typically 4 in realistic situations. For instance, for \( r = 10 \) the power \( p_k \) of the transmitter must be at least 10000.

However, if we introduce a repeater node between transmitter and receiver, say in the middle, we can drastically reduce the whole transmission power. More precisely, to cover the distance of 5 is enough a transmission power of 625. Thus, the transmission power we need for both the transmitter and the repeater is 1250 instead of 10000!

The following result shows that the introduction of a repeater between a first (stationary) node located at some \( l_1 \), and a second (stationary) node located at some \( l_2 \), using a private channel to propagate the signal, does not change the behaviour of the original system. Notice that for \( d(l_1, l_2) = r \), we write \( l_1 + r/2 \) to denote the location placed in the middle, between \( l_1 \) and \( l_2 \).

**Theorem 6.9 (Saving antenna power)** Let \( P \) be a process such that \( \text{fc}(P) = \{d\} \), for some channel \( d \). Let \( l_1, l_2 \) be physical locations, and \( r_1, r_2 \) be transmission radii such that \( d(l_1, l_2) = r \), \( r \leq r_1 \), and \( r \leq r_2 \). Then, the system

\[
(\nu d)(m[P]_{l_1,r/2}^p \mid \text{rr}[d \leftarrow d]_{l_1+r/2,r/2}^p \mid n[P]_{l_2,r_2}^p)
\]

\(^1\)This inequality holds for free-space environments with non-obstructed line of sight, and it does not consider the possible occurrence of reflections, scattering, and diffraction caused by buildings, terrain, and so on. Nevertheless, it is widely accepted in the ad hoc network community.
simulates the system

\[(\nu d)(m[P]_{l_1,r_1} \mid n[Q]_{l_2,r_2}) \]

**Proof** The two systems basically differ for the presence of the range repeater operating on the private channel \(d\). Formally, we prove that the relation

\[
\{(\nu d)(m[P]_{l_1,r_1} \mid n[Q]_{l_2,r_2}), (\nu d)(m[P]_{l_1,r/2} \mid rr[d \leftarrow d]_{l_1+r/2,r/2} \mid n[Q]_{l_2,r_2})
\}

\[
\forall Q \forall P. \text{fc}(P) = \{d\}
\]

is a simulation. □

7 Related and Future Work

Broadcast for Ethernet-like communications has been first analysed by Prasad [20, 21, 16] in his *Calculus of Broadcasting Systems* (CBS), in which all processes receive a broadcast message at once. In [22] the same author proposed a LTS and a (both strong and weak) labelled bisimilarity relying on the notion of “discard relation”, a special transition that any process can perform to discard a potential message. Technically speaking, the discard relation is a mechanism to fit the semantics of broadcast with that of parallel composition.

Hennessy and Rathke [8] proved that the above (weak) bisimilarity, renamed *noisy bisimilarity*, coincides with barbed congruence. Modulo the presence of the discard relation, our bisimilarity is very close to noisy bisimilarity.

The \(b\pi\)-calculus [4] of Ene and Muntean equips the \(\pi\)-calculus with a broadcast paradigm such that only nodes listening on the right channel can receive a broadcast. While this seems to come closer to a notion of local broadcast, it remains complicated to change a once established connectivity. The authors proposed an LTS (relying on the discard relation) and a labelled bisimilarity which is proved to coincide with barbed equivalence. They also proved that the closure under substitution of their labelled bisimilarity corresponds to the barbed congruence.

Nanz and Hankin [15] have introduced a calculus for Mobile Wireless Networks (CBS#) where the recipients of a transmission are determined using a graph representation of node localities. While this approach is more flexible, ours (based on location and radius that define transmission cells and distance) allows a more compact representation of connectivity. The authors
proposed a LTS which is very close to that of [22, 8] and again relies on the
discard relation. This LTS is then used to define a behavioural equivalence,
called mediated equivalence that identifies processes only with respect to their
capability to store items. The final goal of Nanz and Hankin is to use their
calculus as the basis of a framework for specification and security analysis of
communication protocols for MANETs.

Prasad’s more recent calculus of Mobile Broadcasting Systems, (MBS)
[23] aims at providing a communication model which implements the “glob-
ally asynchronous, locally synchronous” communication mechanism which is
proper of wireless communication communication systems. Channels are em-
ployed as sealed rooms, preventing a message sent within a room to being
captured by processes in other rooms.

Singh, Ramakrishnan, and Smolka [26] have proposed the $\omega$-calculus, a
conservative extension of the $\pi$-calculus specifically tailored for modelling
MANETs’ protocols. The key feature of the $\omega$-calculus is the separation of a
node’s communication and computational behaviour from the description of
its physical transmission range. The latter is modelled annotating processes
with the set of group names to which the process belongs. The authors have
proposed a labelled transition semantics that, unlike the previous ones, does
not use the discard relation but instead contains a rule, similar to our (Lose),
to model the non-blocking nature of multicast send. The $\omega$-calculus is then
used for developing a model of the AODV protocol [18], a routing protocol
for MANETs.

Finally, notice that all the previous calculi abstract from interferences.
Mezzetti and Sangiorgi [10] have instead proposed a lower level calculus in
which a node can detect interferences when located in the intersection of
the transmission range of two different nodes. While our syntax is inspired
by that of [10], the reduction semantics and the corresponding LTS is quite
different; this is because in our model we assume the absence of interferences.

**Future Work** A number of developments are possible. For instance, we
could enrich the calculus with operators to model the concept of store as in
[15]. We could try to extend the behavioural theory to deal with node failure.
At this regards, the developments in [5, 6] for wired networks could be a good
starting point. Moreover, wireless systems have also features of synchrony
that remind us of synchronous languages (e.g. Esterel [3], Statecharts [7],
SCCS [11]). Indeed, in a single time unit of a wireless system multiple events
can happen. It is our intention to investigate these aspects taking inspiration
from [23]. Finally, as pointed out in [15], security is, of course, another
important issue in MANETs that we would like to investigate.
Acknowledgements  Many thanks to Davide Sangiorgi for insightful comments on a early draft. We also thank Davide Quaglia for interesting discussions on MANETs.

References


REFERENCES


