Journal of Information Processing Vol. 19 364–377 (July 2011)

Regular Paper

Performance Evaluation of Personal and General Data Classes for Trust Management in MANETs

Marcin Seredynski,[†1,†2] Pascal Bouvry[†1] and Dominic Dunlop[†1]

The necessary cooperation in packet forwarding by wireless mobile ad hoc network users can be achieved if nodes create a distributed cooperation enforcement mechanism. One of the most significant roles in this mechanism is played by a trust system, which enables forwarding nodes to distinguish between cooperative (therefore trustworthy) and selfish (untrustworthy) nodes. As shown in this paper, the performance of the system depends on the data classes describing the forwarding behaviour of nodes, which are used for the evaluation of their level of cooperation. The paper demonstrates that partition of such data into personal and general classes can help to create better protection against clique-building among nodes. Personal data takes into account the status of packets originated by a node itself, while general considers the status of packets originated by other nodes. Computational experiments demonstrate that, in the presence of a large number of selfish and colluding nodes, prioritising the personal data improves the performance of cooperative nodes and creates a better defence against colluding free-riders.

1. Introduction

Mobile ad hoc networks (MANETs) are a class of dynamic multi-hop networks composed of a set of mobile nodes that can communicate using shared channels without support from any fixed infrastructure [1]. As nodes can move, the topologies of such networks change dynamically. Packet delivery is based on multi-hop routing, which means that nodes are expected to act as both terminals and routers. In contrast to traditional networks, network users in civilian MANETs may belong to different authorities; thus one cannot assume that routers (nodes) are trustworthy (i.e., perform routing functionality correctly) [2]. Moreover, among a node's salient characteristics is energy-constrained operation (as most of them rely on batteries) [3]. Consequently, the risk of selfish behaviour (discarding packets received for forwarding) driven by the need for energy conservation need is very high. Therefore civilian MANETs will most probably suffer from free-riding behaviour [4)–6)] unless distributed *cooperation enforcement mechanisms* (CEMs) are created by participating nodes. The goal of a CEM is to provide incentives for cooperation, making it a rational choice (leading nodes to the maximisation of their benefits [7]). Such mechanisms fall into two categories: *pricing* and *trust/reputation-based* [8]. The pricing-based approach can be seen as an economic view of the problem. The general idea is that nodes have to pay for receiving service and are paid for providing it. The work presented in this paper is limited to the second category — trust/reputation-based mechanisms, where the general idea is that intermediate nodes forward packets only on behalf of cooperative (therefore trustworthy) nodes. Several trust-based cooperation enforcement mechanisms have been proposed in the literature (e.g., Refs. 9)–17)). The key component of such mechanisms is a *trust system* that enables cooperative nodes to be distinguished from selfish. The trust system is often extended with a reputation system. The main difference between the two is that, in the former, a node evaluates a subjective view of the entity's level of cooperation, while in the latter the view of the whole community is taken into account [18]. Trust/reputation management in the context of cooperation enforcement mechanisms can be seen as a *soft security* approach (similar to the *social control mechanisms* [19] used in Internet commerce). Its goal is to create a *strategy-proof* network, i.e., a network that is resistant to the strategic behaviour of *selfish* (but not malicious) users wanting to exploit it [20]. This differs from the goal of hard security mechanisms, which is to provide protection against malicious nodes [18]. This paper deals only with the problem of selfish nodes.

As cooperation in MANETs involves human behavioural and social factors [21], the underlying history-based mechanisms of CEMs are *direct* and *indirect reciprocity*. Direct reciprocity-based cooperation can be characterised as "I forward your packets and you will reciprocate in the future by forwarding mine", while indirect reciprocity is "I forward your packets, and somebody else will indirectly reciprocate by forwarding mine". The former requires nodes to memorise their

†1 Faculty of Sciences, Technology and Communication, University of Luxembourg
†2 Interdisciplinary Centre for Security, Reliability and Trust, University of Luxembourg

bilateral forwarding interactions, while the latter expects them to track interactions between other nodes.

In CEMs found in the literature, data used for the evaluation of cooperation level (herein referred to as *trust data*) is classified as either *private* (first-hand observations) or *public* (second-hand observations obtained from third parties). We introduced a further distinction between *personal* and *general data* in a previous paper [22]. The former takes into account the status of packets originated by a node itself, while in the latter the status of packets originated by other nodes is considered. Both classes of data can result from first- or second-hand observations. As demonstrated in this paper, one of the situations where such a distinction might be significant is when some nodes build a clique by being cooperative only within its membership. If general trust data are used, other nodes interpret such behaviour as cooperative in spite of the fact that it is only cooperative to the members of the colluding group. Consequently, by creating a clique, selfish nodes might be able to obtain similar performance to cooperative ones.

The main contribution of this paper is the analysis of the distinction between personal and general data classes in the context of clique building among selfish nodes. The simulation-based study was performed using the trust-based packet forwarding introduced in Ref. 17). However, the implications of the study are valid for any trust/reputation-based CEM.

The paper is structured as follows. The next section provides a survey of related work. Section 3 addresses the network assumptions and describes the classification of trust data classes in the context of trust systems used in MANETs. Section 4 explains the model of the network used to analyse the significance of data distinction in the presence of clique building. Section 5 contains a description of the experimental design and simulation results. The final section summarises the main conclusion.

2. Background and Related Work

Surveys of trust and reputation systems for cooperation enforcement in MANETs can be found in Refs. 21), 23). The most closely-related work is concerned with the classification of data classes for the evaluation of the cooperation level of nodes. A distinction is made between first- and second-hand information. In general, first-hand observations are more reliable than second-hand [18]. The question of whether to use second-hand information or not is basically related to the trade-off between the speed of the evaluation of the level of cooperation and the robustness of such an evaluation [24]. In Ref. 10), a cooperation enforcement mechanism called CORE is presented. According to the proposal, the level of cooperation is evaluated using first- and second-hand evaluations (both having the same significance). However, the second-hand ratings include only information about cooperative behaviour. Consequently, the possibility of the malicious broadcast of negative ratings for legitimate nodes is avoided. The reliability of the trust evaluation is also positively correlated with the number of evaluations taken into account and its variance. In Ref. 9) the authors propose a protocol called *CONFIDANT*, where negative second-hand rating is allowed. However, a node's own experience is rated higher than second-hand reports. In Refs. 11), 25) the use of second-hand information is further investigated. A Bayesian approach is introduced: opinions that deviate from the first-hand observations and from the opinion of the majority are excluded. As a result, the reputation system is much more robust against false accusations and benefits from a faster detection of selfish nodes. In Ref. 24) the authors apply the mean-field approach to a proposed stochastic process model to demonstrate that liars have no impact unless their number exceeds a certain threshold. In the SORI algorithm of Ref. 12), ratings are only exchanged between neighbours. The level of cooperation of the rater is positively correlated with its ratio of packets forwarded to packets discarded on behalf of the evaluator. In Ref. 22) trust data are classified into two classes referred to as *personal* and *general*. The former considers the status of packets originated by a node itself, while in the latter, the status of packets originated by other nodes is taken into account (more details regarding the distinction will be given in Section 3.2). The authors do not, however, provide an experimental evaluation of the influence of these classes on the performance of nodes. In Ref. 26) the authors demonstrate that, if cooperation is based on indirect reciprocity and a classic watchdog-based mechanism [5] is used for data collection, discarding packets can be seen as an act of altruistic punishment. In such a situation an intermediate node that discards packets from selfish senders pays a

cost expressed as a decrease of the level of cooperation among other nodes. If the cost of punishing free-riders is too high, then nobody has an incentive to be the punisher. A direct reciprocity-based cooperation with several forwarding strategies present in the network is analyzed in Refs. 14), 16), 27). In these works the authors demonstrate that cooperation is very likely to be developed on the basis of defection-tolerant versions of the reciprocal tit-for-tat approach. In MANETs, trust systems have also been combined with routing protocols in order to bypass misbehaving nodes (see e.g., Refs. 2), 5)).

3. Network Assumptions and Trust Data

This section presents network assumptions, and describes the watchdog-based mechanism for collection of personal and general trust data.

3.1 Network Assumptions

The behaviour of nodes in a MANET is affected by many factors like the number and heterogeneity of network participants, *a priori* trust relationships, monitoring techniques, etc. In this paper the following assumptions about the network are made:

- the network is self-organising and the network layer is based on a reactive, source routing protocol;
- the topology of the network is unpredictable and changes dynamically;
- each device is equipped with an omnidirectional antenna with similar radio range, bi-directional communications and promiscuous mode;
- network users pursue their own self-interest;
- the evaluation of a cooperation level of a given node is relative to behaviour of others, i.e., nodes maintain subjective rankings of other network participants according to their forwarding behaviour;
- the decision whether to accept or reject a forwarding request is probabilistic and positively correlated with the level of cooperation of the source of the packet.

3.2 Local Trust System: Personal and General Data Classes

In this article a distinction between personal and general trust data is made using the example of a *watchdog mechanism* [5]. The observable elements used to derive the level of cooperation of the source of the message are two network events:

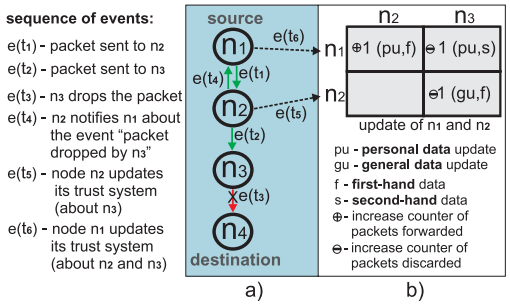

Fig. 1 Example of the watchdog-based trust data collection mechanism: communication session between nodes n_1 and n_4 failed because the packet was discarded by node n_3 (a), trust data update after the communication session (b).

"packet forwarded" and "packet discarded". As a source routing protocol is used, a list of intermediate nodes is included in the header of the packet. Information regarding the packet forwarding behaviour of other nodes (trust data) is gathered only by nodes directly participating in the *communication session*. There is no exchange of ratings between nodes. The communication session involves a source node (sender), several forwarders (nodes that forward packets) and a destination node. Trust data collection is performed in the following way: nodes are equipped with a *watchdog mechanism* that enables them to check whether a packet was delivered to its destination. A node that requests another node to forward a packet verifies by means of a *passive acknowledgement* [28] whether the requested node actually forwarded the packet. As an example, let us assume that node n_1 originates a message to node n_4 via intermediate nodes n_2 and n_3, and the message is eventually discarded by node n_3 (**Fig. 1** a). This event is recorded by the watchdog mechanism of node n_2, which next informs n_1 about the selfish behaviour of n_3. As a result, the trust system of node n_1 is updated with two events — "packet forwarded by n_2" and "packet discarded by n_3", while the trust system of n_2 is updated with the event "packet discarded by n_3" (Fig. 1 b). Such a watchdog-based mechanism has certain vulnerabilities (see Refs. 2), 5)); however, the proposed distinction between personal and general trust data is not limited only to this particular approach for data collection.

Trust data can be classified into two classes referred to as *personal* and *gen-*

Table 1 Modes of the trust system and related mechanisms for cooperation mechanisms.

	data classes used	**underlying mechanisms**
G mode	general	indirect reciprocity, altruistic punishment
P mode	personal	direct reciprocity
PG mode	personal and general	direct and indirect reciprocity, altruistic punishment
PPR mode	personal and if personal unavailable then general	direct reciprocity, limited indirect reciprocity and altruistic punishment

eral [22]. The former considers the status of packets originated by a node itself, while in the latter the status of packets originated by other nodes is taken into account. Thus, a node in a sender role collects personal trust data, while forwarders collect general trust data. Both classes of data can be obtained by either first- or second-hand observations. The type of data used to evaluate the cooperation level of a sender (so providing a basis for a decision on forwarding) results in the development of particular mechanisms underlying cooperation. If such a decision is made on the basis of personal trust data only, cooperation is built using direct reciprocity. The mechanism assumes one-on-one interactions between the sender and forwarders. On the other hand, the use of general trust data opens a possibility for the development of cooperation on the basis of an indirect reciprocity mechanism. If some nodes decide to collude within a group (by being cooperative only to its members) this will have an influence on general trust data, as the non-colluding nodes interpret the colluding behaviour as cooperative (when in fact it is only cooperative to selected nodes).

On the basis of the distinction between personal and general trust data, four modes of the trust system are defined (see **Table 1**). In the first, denoted G, a node evaluates the cooperation level of others using only general trust data. In the second (P), it uses only personal trust data. In the third (PG), it uses both data classes (with no distinction between them being made). In the final mode (PPR), personal data are preferred over general, i.e., a node uses personal data; however, when it is unavailable general data are taken into account.

4. Trust-based Forwarding Approach

The influence of personal and general trust data on the performance of a trust system was evaluated in this work using a cooperation enforcement mechanism proposed in Ref. 17). According to this mechanism the evaluation of cooperation level of a given node is relative to behaviour of others. Thus, nodes, instead of being classified as selfish or cooperative according to some fixed values, are compared against each other, resulting in their subjective ranking (according to the level of cooperation). Such an approach helps to deal with the uncertainty (due mainly to the mobility of nodes) related to the trust data collection mechanism. The approach assumes that possible false feedback affects all nodes in the same way, and thus does not modify the ranking of nodes.

The mechanism assumes that each node uses two components, a local *trust evaluation* algorithm and a *probabilistic forwarding strategy*. Nodes collect data about the forwarding behaviour of other network participants according to the scheme described in Section 3.2. Each time an intermediate node receives a *forwarding request* (FREQ) it checks the cooperation level of the original sender (source node) of the message. The decision as to whether to relay or discard the packet is specified by the probabilistic forwarding strategy, which defines the probability at which the packet is forwarded. This probability is positively correlated with cooperation level of the source.

The evaluation of the *cooperation level* of node j (source of the packet) by node i asked to relay the packet is based on two characteristics: *relative forwarding rate* (denoted by $rfr_{j|i}$) and *relative activity* (denoted by $rac_{j|i}$). The notation $j|i$ means that node j is under evaluation by node i. Both values result from the watchdog-based observations made by node i during its participation in the network. The calculation of relative forwarding rate is based on the ratio of packets forwarded to packets discarded by a node, while the calculation of relative activity depends on the node's degree of participation to the forwarding duties. Node i observes two characteristics of j, namely $npf_{j|i}$ and $npd_{j|i}$, which are

respectively the numbers of packets forwarded and discarded by j. On the basis of these characteristics the node computes a *FREQ acceptance ratio* ($rar_{j|i}$) for node j:

$$rar_{j|i} = \frac{npf_{j|i}}{npf_{j|i} + npd_{j|i}}. \quad (1)$$

The values $rfr_{j|i}$ and $rac_{j|i}$ are functions of $rar_{j|i}$ and $npf_{j|i}$ respectively:

$$rfr_{j|i} = \frac{1}{||\mathbb{O}_i|| - 1} \cdot \sum_{k \in \mathbb{O}_i} \langle rar_{k|i} < rar_{j|i} \rangle, \quad (2)$$

$$rac_{j|i} = \frac{1}{||\mathbb{O}_i|| - 1} \cdot \sum_{k \in \mathbb{O}_i} \langle npf_{k|i} < npf_{j|i} \rangle, \quad (3)$$

where

$$\langle Statement \rangle = \begin{cases} 1 & \text{if } Statement \text{ is true,} \\ 0 & \text{if } Statement \text{ is false.} \end{cases}$$

$\mathbb{O}_i$ denotes a set of all nodes observed by node i and $||\mathbb{O}_i||$ stands for its size. The relative forwarding rate value defined by Eq. (2) is the fraction of nodes that have a lower FREQ acceptance ratio than node j. A similar interpretation holds for the relative activity.

In the next step node i evaluates the cooperation level of node j ($ev_{j|i}$), which is a number from the interval $[0, 1]$. It is defined as a function of relative forwarding rate and relative activity:

$$ev_{j|i} = f_i(rfr_{j|i}, rac_{j|i}), \quad (4)$$

where $f_i : [0, 1] \otimes [0, 1] \to [0, 1]$.

Finally, $ev_{j|i}$ is computed as the mean of the relative forwarding rate and activity:

$$ev_{j|i} = \frac{rfr_{j|i} + rac_{j|i}}{2}. \quad (5)$$

The next step for a node that receives a packet for forwarding is to decide whether to accept or reject it. The decision is specified by the forwarding strategy, which is a function of cooperation level, and is denoted by $s_i(ev_{j|i})$. A probabilistic strategy is used, i.e., $s_i(ev_{j|i})$ is the probability that i will relay a packet originated by j. Such a probability is defined by the cumulative distribution function

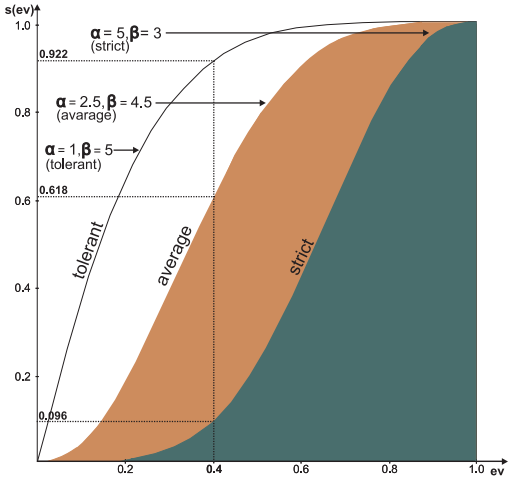

Fig. 2 An example of three probabilistic forwarding strategies.

of the beta distribution:

$$s_i(ev_{j|i}) = \int_0^{ev_{j|i}} d_{\alpha,\beta}(x)dx, \quad (6)$$

where $d_{\alpha,\beta}(x)$ is the *density function* of the beta distribution with parameters α and β, i.e., for $x \in [0, 1]$,

$$d_{\alpha,\beta}(x) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{\int_0^1 u^{\alpha-1}(1-u)^{\beta-1}du}. \quad (7)$$

Parameters α and β determine the shape of $s_i(x)$. The advantage of the beta distribution is that changing two parameters enables coverage of a large set of strategies differing in their level of cooperativeness. An example of three forwarding strategies differing in their strictness is shown in **Fig. 2**. The three strategies denoted by *strict*, *average*, and *tolerant* differ in the values of their α and β parameters. As an example, let us assume that the cooperation level of the source of the packet is equal to 0.4. This means that the packet is forwarded with a probability around 0.1 (if the strict strategy is used), 0.62 (if the average strategy is used) and 0.92 (if the tolerant strategy is used).

Two cases concerning messages originated by unknown nodes are specified. If

FREQ occurs in the initial period of the existence of the network, the packet is forwarded with a probability p_1. Otherwise, the packet is forwarded with a probability p_2. In general, p_1 is high (to let newcomers integrate with the network), while p_2 is low (to discourage network participants from whitewashing behaviour [29], i.e., changing identity in order to take advantage of the cooperative approach to unknown nodes). The initial period of the existence of the network is specified by a threshold parameter t_{unkn} (time until which the preferential p_1 probability is used). Trust data are also used by nodes for sending their own packets to rate the available paths to the destination. The rating is calculated as the arithmetic mean of cooperation levels of all nodes belonging to the route. The path with the best rating is chosen.

5. Experiments

The goal of the experiments performed was to analyse the significance of the partitioning of data into personal and general classes in the presence of a clique of selfish nodes. Four modes of the trust system (defined in Section 3.2) differing in the use of data classes were tested in various network settings. These settings differed in the composition of types of nodes (the type of a node specifies its forwarding approach and the time at which the node enters the network). Two forwarding approaches were defined. The first one is the *cooperative* (*trust-based*) approach (denoted by TB) explained in Section 4. The second one is a *non-cooperative* (*selfish*) approach (denoted by NC). Three types of nodes were specified: (i) nodes that were present in the network from its beginning and used the TB approach (these nodes are referred to as *C-type*), (ii) nodes that used the same approach but appeared in the network at some later stage (nodes referred to as *CL-type*) and (iii) selfish nodes (referred to as *N-type*) that were in the network from the beginning and used the NC forwarding approach. The N-type nodes forwarded packets with a probability of 0.1. However, these nodes built a clique by forwarding all packets on behalf of all nodes of the same type (it was assumed that N-type nodes recognised themselves even without having any prior direct or indirect interactions). The parameter specifications of the common settings of all sets of experiments are given in **Table 2**.

Table 2 Specification of the parameters.

Parameter	Value
# of all nodes (M)	60
simulation time (R)	600
α of TB approach	2.5
β of TB approach	4.5
p_1 of TB approach	1.0
p_2 of TB approach	0.3
t_{unkn} of TB approach	round no.50
cooperation level of an unknown node in path rating mechanism	0.3
forwarding probability of NC approach	0.1 (to C/CL) 1.0 (to N)
round number at which CL-type nodes join the network	200
path length (number of hops)/ probability	1h/0.1, 2h/0.3, 3–5h/0.2
number of available paths	1–4 (equiprobable)

The total number of nodes was 60, while the simulation time was set to 600 rounds (the simulation procedure will be described in Section 5.1). The path length ranged from 1 up to 5 hops with the following probabilities: one hop – 0.1, two hops – 0.3 and three to five hops – 0.2. The number of available paths from a source to a given destination ranged from 1 to 4 (each value equiprobable). The following performance measures of a node belonging to a given type were used: a *throughput*, defined as a ratio of successfully delivered messages; a *number of packets forwarded* (npf), reflecting contribution to forwarding duty; and *forwarding rates* (fr), specifying a ratio of packets forwarded to discarded (on behalf of all nodes and on behalf of nodes of a given type). Each experiment was repeated 100 times. The value of each performance measure was calculated as the mean value of the performance of a single node belonging to a given type over all runs of a given experiment. The performance of each mode of the trust system was evaluated as a function of the number of selfish (N-type) nodes present in the network. As for the remaining (non-selfish) nodes, about 90% of them were of C-type and 10% of CL-type.

5.1 Experimental Procedure

The network was composed of a finite population of M nodes, while time was divided into R *rounds*. In each round every node initiated a single communication session exactly once (acted as a sender), thus each round was composed of M

communication sessions. As the dynamics of a typical MANET expressed in terms of mobility and connectivity of the nodes are unpredictable, intermediate and destination nodes were chosen randomly. The simulation of the network for a given experiment was defined by the following algorithm:

(1) Specify i (source node) as $i := 1$, M as a number of nodes participating in the network and R as a number of rounds and next set strategies of nodes as specified by the experiment;
(2) Randomly select node j (destination of the packet) and intermediate nodes, forming several (exact number specified in Table 2) possible paths from node i to j;
(3) If more than one path is available, calculate the rating of each path and choose the path with the best rating (as described in Section 4);
(4) Let node $i := 1$ initiate a communication session (originate a packet). The packet is next either passed on or dropped by intermediate nodes according to their forwarding approaches;
(5) After the completion of the communication session, update trust data (as described in Section 3.2);
(6) If $i < M$, then choose the next node ($i := i + 1$) and go to step 2. Else go to step 7;
(7) If $r < R$, then $r := r + 1$ and go to step 1 (next round). Else, stop the simulation.

All nodes communicate exactly one message in each round. This simplification corresponds to a situation where all network users send a similar number of packets. We experimentally verified that if nodes originated a random number of packets (from a similar range), this would not have any significant influence on the results.

5.2 Results and Discussion

The performance of CL-type nodes as a function of the number of selfish (N-type) nodes present in the network is shown in **Fig. 3** (throughput) and **Fig. 4** (contribution to packet forwarding). Detailed numerical values for three selected cases (no N-type nodes in the network, 18 N-type, and 36 N-type) are shown in **Table 3**.

In general, the throughputs ranged from 0.40–0.56 (when no selfish nodes were present in the network) to 0.21–0.29 (when 36 of 60 nodes were selfish). As soon as 15 to 22 nodes were of N-type, very similar throughputs were observed in each mode of the trust system, however, their contribution to packet forwarding was lower in P and PPR modes (Fig. 4). Otherwise, the best performance (highest throughput and lowest contribution to packet forwarding) was achieved when

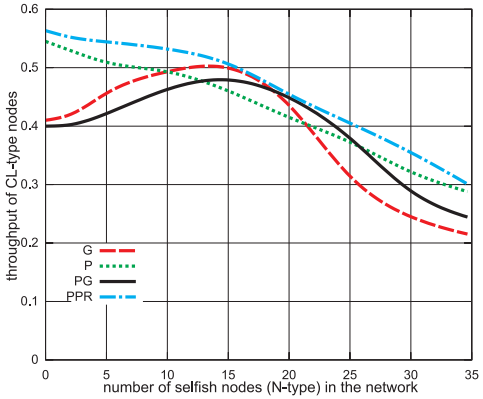

Fig. 3 Throughput of CL-type nodes as a function of the number of selfish nodes present in the network.

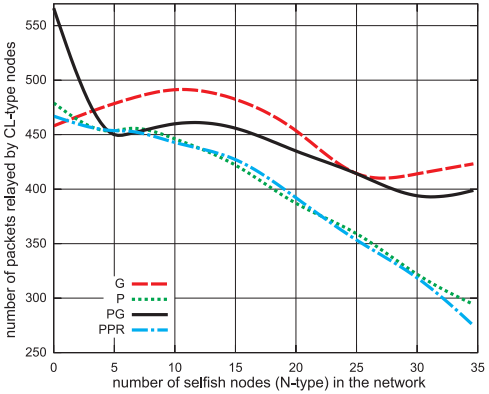

Fig. 4 Contribution to packet forwarding of CL-type nodes as a function of the number of selfish nodes present in the network.

Table 3 Average performance of a single node belonging to CL, C or N-type class in three cases: (i) no N-type present in the network, (ii) 18 N-type nodes in the network and (iii) 36 N-type nodes in the network. Abbreviations: fr- forwarding rate, npf - number of forwarded packets.

	0 N-type				**18 N-type**				**36 N-type**			
	G	P	PG	PPR	G	P	PG	PPR	G	P	PG	PPR
throughputs of CL nodes	0.41	0.55	0.40	0.56	0.46	0.43	0.47	0.48	0.21	0.28	0.24	0.29
npf by CL	459	479	437	467	467	400	444	404	426	288	403	261
fr of CL vs. all	0.79	0.76	0.77	0.76	0.80	0.68	0.77	0.70	0.81	0.52	0.77	0.49
fr of CL vs. N only	-	-	-	-	0.38	0.34	0.30	0.30	0.85	0.33	0.76	0.30
throughputs of C nodes	0.68	0.71	0.67	0.70	0.57	0.58	0.59	0.61	0.23	0.32	0.29	0.34
npf by C	854	888	830	883	771	746	772	786	652	469	657	489
fr of C vs. all	0.81	0.76	0.80	0.83	0.79	0.77	0.78	0.79	0.80	0.56	0.78	0.58
fr of C vs. N only	-	-	-	-	0.34	0.27	0.27	0.25	0.83	0.33	0.73	0.34
throughputs of N nodes	-	-	-	-	0.42	0.39	0.39	0.38	0.90	0.65	0.84	0.66
npf by N for all	-	-	-	-	364	335	364	364	863	742	844	756
npf by N for C/CL only	-	-	-	-	52	50	51	50	28	25	27	25

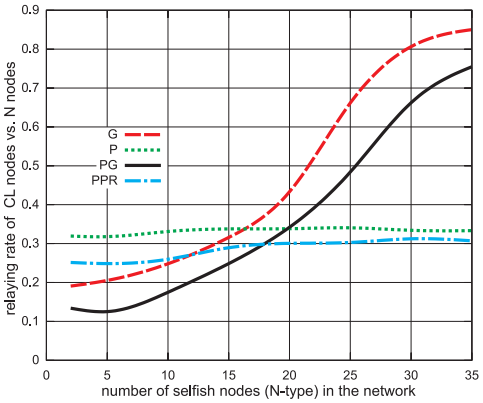

Fig. 5 Forwarding rates of CL nodes vs. packets received from selfish nodes.

nodes evaluated the cooperation level of others using PPR mode. The P mode was only slightly behind. The worst performance was observed when G mode was used (except that, when fewer than 18 nodes were of N-type, G mode provided slightly greater throughput than PG mode). On average, the best defence against the selfish behaviour of N-type nodes was created by PPR mode (forwarding rate to packets from N-type nodes was in 0.21–0.30 range, see **Fig. 5**). However, whenever fewer than 18 selfish strategies were present, the best strategy-proof network was created by PG mode (with forwarding rate ranging from 0.12 to 0.30). With a large presence of selfish nodes, defences created by PG and G modes against selfish nodes were very poor. Nodes using these modes forwarded up to 76% (PG) and 85% (G) packets on behalf of selfish nodes. On the other hand, the highest forwarding rates of P and PPR modes were 0.33 and 0.30 respectively.

Throughputs of C-type nodes are shown in **Fig. 6**. Detailed numerical values for three selected cases presented in Table 3. These nodes obtained higher throughputs compared to CL-type (0.68–0.71 when no selfish nodes were present and 0.23–0.34 with 36 N-type nodes). Again, nodes that used the PPR mode obtained the best throughputs. The exception was when selfish nodes were not present in the network. In such a case, the P mode was slightly better (0.71 vs. 0.70). The contribution to packet forwarding is shown in **Fig. 7**, while forwarding rates to packets from selfish nodes are demonstrated in **Fig. 8**. Obviously, the C-type nodes forwarded more packets than CL-type. This was due to the fact that these nodes were present in the network from its beginning, while CL-type nodes joined the network in round 200. In terms of differences between modes of the trust system and achieved performance, similar observations to those for the case of CL-type nodes can be made. However, there are three additional remarks. Firstly, the differences in terms of throughputs and contributions to packet for-

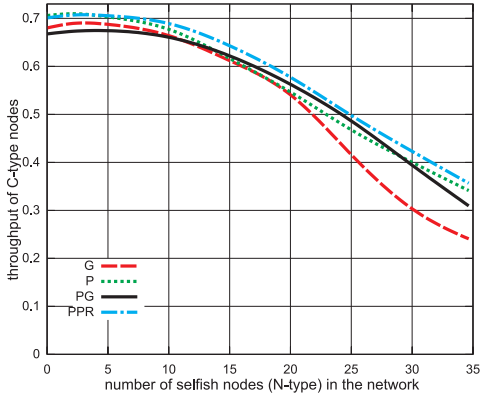

Fig. 6 Throughput of C-type nodes as a function of the number of selfish nodes present in the network.

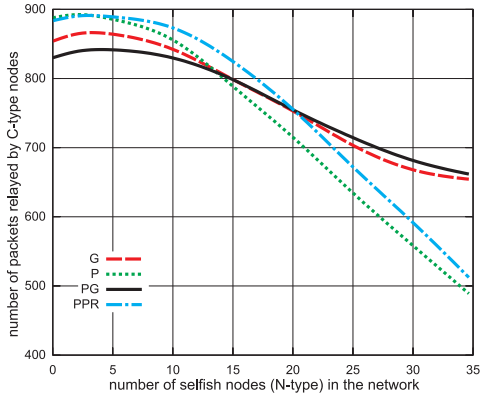

Fig. 7 Contribution to packet forwarding of C-type nodes as a function of the number of selfish nodes present in the network.

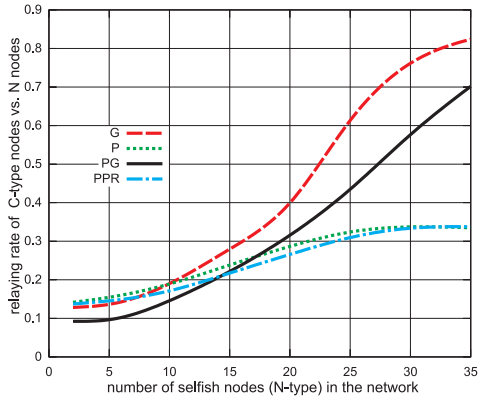

Fig. 8 Forwarding rates of C nodes vs. packets received from selfish nodes.

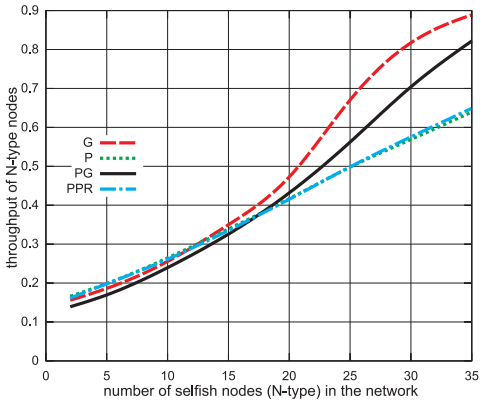

Fig. 9 Throughput of N-type nodes as a function of the number of selfish nodes present in the network.

warding were smaller than those observed in the case of CL-type nodes. Secondly, nodes in P mode almost always forwarded less packets than the ones in PPR mode (see Fig. 7). In contrast, in the case of CL nodes, PPR and P modes achieved similar performances in terms of contribution to packet forwarding (see Fig. 4). Lastly, C-type nodes created better defences against selfish nodes than CL-type nodes, especially in the presence of a small number of selfish nodes (compare Fig. 5 and Fig. 8).

The performance of the last type of node (selfish N-type) is shown in **Fig. 9** (throughput) and **Fig. 10** (contribution to packet forwarding). Detailed numerical values for three selected cases are provided in Table 3. Initially (when up to around 18 N-type nodes were present in the network) the P and PPR modes of C and CL-type nodes slightly favoured selfish nodes (compared to the remaining

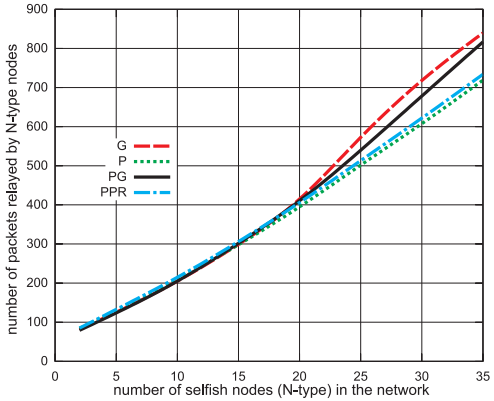

Fig. 10 Contribution to packet forwarding of N-type nodes as a function of the number of selfish nodes present in the network.

modes). However, from that point on, C and CL-type nodes that used G and PG modes became more and more intolerant towards selfish nodes (see Fig. 5 and Fig. 8). The significant packet forwarding contribution of N-type nodes was mainly oriented to requests from other N-type nodes (see Fig. 10). These nodes forwarded from 335 to 863 packets on behalf of the members of the clique and only 25–52 for the remaining types (see Table 3).

In the experiments described so far in this section, it was assumed that the dynamics of the network were unpredictable. Therefore, intermediate and destination nodes were chosen randomly. We now address the question of what would happen if some correlation was introduced (resulting for instance from lower mobility). This was addressed in the following way in the simulations. The network was divided into three zones of equal area. Nodes could interact only with those which were in the same zone. After every time step (modelled by a round), 10% of the nodes changed their zones. The comparison of the previous results with those obtained with zoning is shown in **Table 4**.

In general, PPR remained the best mode. However, a few minor changes were observed. When N-type nodes were not present, PPR replaced P as the best mode for C-type nodes and PG moved up one place in the ranking, leaving G as the worst mode. When 18 N-type nodes were present in the network, the PG

Table 4 Comparison of throughputs of nodes in two cases: (i) intermediate and destination nodes chosen over the whole network, (ii) intermediate and destination nodes chosen within a zone.

	0 N-type				**18 N-type**				**36 N-type**			
	G	P	PG	PPR	G	P	PG	PPR	G	P	PG	PPR
(i) CL	0.41	0.55	0.40	0.56	0.46	0.43	0.47	0.48	0.21	0.28	0.24	0.29
(ii) CL	0.50	0.56	0.43	0.58	0.47	0.44	0.48	0.48	0.24	0.30	0.25	0.31
(i) C	0.68	0.71	0.67	0.70	0.57	0.58	0.59	0.61	0.23	0.32	0.29	0.34
(ii) C	0.63	0.69	0.67	0.70	0.51	0.57	0.59	0.60	0.23	0.35	0.31	0.36
(i) N	-	-	-	-	0.42	0.39	0.39	0.38	0.90	0.65	0.84	0.66
(ii) N	-	-	-	-	0.48	0.41	0.42	0.41	0.90	0.67	0.84	0.68

Table 5 Comparison of throughputs of nodes in three cases differing in the strategies used by C and CL-type nodes: (i) tolerant strategy, (ii) average strategy, (iii) strict strategy.

	0 N-type				**18 N-type**				**36 N-type**			
	G	P	PG	PPR	G	P	PG	PPR	G	P	PG	PPR
(i) CL	0.81	0.77	0.70	0.84	0.58	0.56	0.60	0.62	0.29	0.33	0.32	0.36
(ii) CL	0.41	0.55	0.40	0.56	0.46	0.43	0.47	0.48	0.21	0.28	0.24	0.29
(iii) CL	0.22	0.35	0.22	0.32	0.20	0.30	0.22	0.28	0.15	0.26	0.15	0.23
(i) C	0.94	0.93	0.92	0.94	0.63	0.63	0.64	0.66	0.30	0.36	0.34	0.37
(ii) C	0.68	0.71	0.67	0.70	0.57	0.58	0.59	0.61	0.23	0.32	0.29	0.34
(iii) C	0.35	0.40	0.37	0.40	0.30	0.42	0.38	0.40	0.16	0.31	0.16	0.27
(i) N	-	-	-	-	0.42	0.32	0.32	0.34	0.82	0.61	0.77	0.62
(ii) N	-	-	-	-	0.42	0.39	0.39	0.38	0.90	0.65	0.84	0.66
(ii) N	-	-	-	-	0.66	0.58	0.65	0.58	0.29	0.33	0.32	0.36

mode performed as well as the PPR. The greatest differences were observed in the case of the G mode: in the 0 N-type network, they improved their throughput by 9% (for CL-type nodes) and worsened by 5% for C-type nodes. As for the other modes, the biggest changes were observed in the 36 N-type network, as C- and CL-type nodes improved their throughputs by 1 to 3%. Selfish nodes obtained slightly better throughput (1–2% higher, except that when 18 N-type nodes were present and G mode was used, the increase was 6%).

In the experiments reported so far, C- and CL-type nodes used the same forwarding strategy (with α equal to 2.5 and β equal to 4.5). The question arises as to what would happen if different strategies were used. To address this issue, two additional strategies (the strict and the tolerant strategy) were introduced. The first (α equal to 5 and β equal to 3) forwards packets with a lower probability compared to the strategy used so far (referred to as the average strategy). On

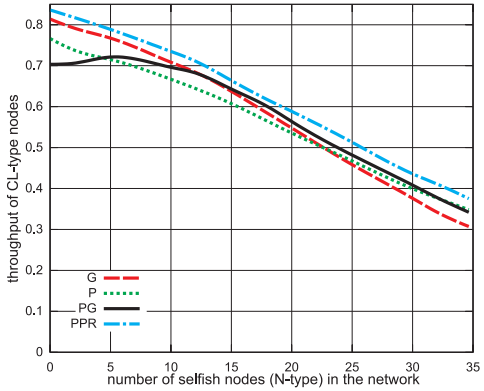

Fig. 11 Throughput of CL-type nodes using a tolerant forwarding strategy.

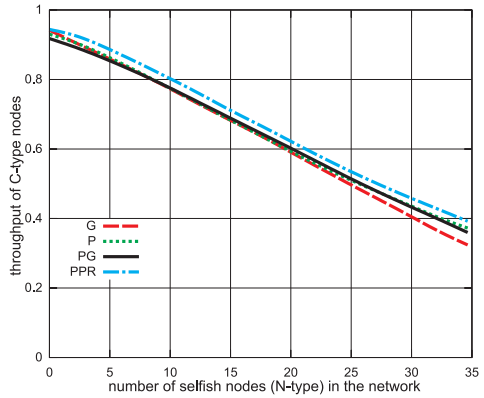

Fig. 12 Throughput of C-type nodes using a tolerant forwarding strategy.

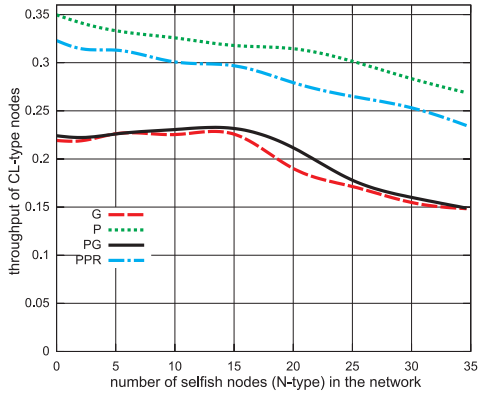

Fig. 13 Throughput of CL-type nodes using a strict forwarding strategy.

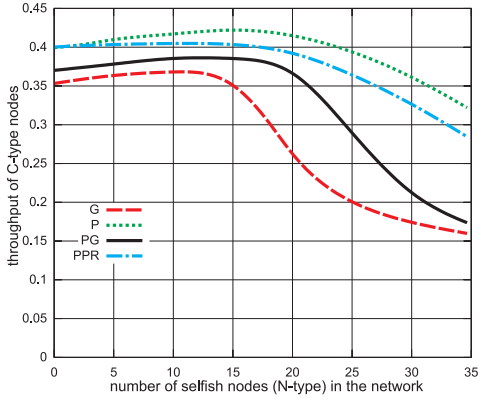

Fig. 14 Throughput of C-type nodes using a strict forwarding strategy.

the other hand, the tolerant strategy is a relaxed version of the average strategy (its α parameter value is set to 1 and β set to 5). These strategies are compared in Fig. 2. The experiments were re-run with the new strategies. Detailed results comparing the performances of the three strategies are shown in **Table 5**.

The performances of nodes using a more tolerant version of the forwarding strategy are shown in **Fig. 11** (CL-type nodes) and **Fig. 12** (C-type nodes). The performances of nodes using a stricter version of the forwarding strategy are shown in **Fig. 13** (CL-type nodes) and **Fig. 14** (C-type nodes). **Table 6** compares the contribution to packet forwarding of the three strategies.

When the tolerant forwarding strategy was used, the differences between the modes were lower than before. Nevertheless, the best throughput was achieved by the PPR mode (as in the case of the average strategy). In contrast, the use of the strict strategy resulted in greater differences between the P/PPR and the PG/G modes (in favour of P/PPR). In addition, throughputs of nodes that used

Table 6 Comparison of contribution to packet forwarding (number of forwarded packets) in three cases differing in the strategies used by C and CL-type nodes: (i) tolerant strategy, (ii) average strategy, (iii) strict strategy.

	0 N-type				**18 N-type**				**36 N-type**			
	G	P	PG	PPR	G	P	PG	PPR	G	P	PG	PPR
(i) CL	637	600	607	595	575	551	570	549	504	444	507	426
(ii) CL	459	479	437	467	467	400	444	404	426	288	403	261
(iii) CL	253	300	243	257	257	276	250	231	304	250	249	188
(i) C	1197	1180	1159	1206	974	923	975	963	841	668	851	698
(ii) C	854	888	830	883	771	746	772	786	652	469	657	489
(iii) C	400	460	417	458	387	490	433	461	433	388	371	335

the P mode were 2 to 4% higher than those for the PPR mode. In both cases (the strict and the tolerant strategy) the lowest contribution to packet forwarding in the presence of a large number of selfish nodes was observed when the nodes used the P and PPR modes.

6. Conclusion

The problem of lack of cooperation on packet forwarding is one of the most important soft security issues in a civilian application MANET. Relaying packets is rational from a node's point of view only if it is positively correlated with the service received from the network. Such a correlation can be obtained if nodes interact only with those they find trustworthy due to their cooperative behaviour in the past. However, the question remains, what kind of information regarding the behaviour of nodes should be used to evaluate the cooperation level before passing on a packet to the next hop. In this paper we have demonstrated that a distinction between personal (related to direct reciprocity) and general (related to indirect reciprocity) trust data is important in the presence of a large number of colluding nodes. We have demonstrated that in a network where some nodes belong to a clique which is cooperative within its membership and selfish otherwise, the best throughput for remaining nodes can be obtained when personal data are favoured over general. However, if nodes are using very strict forwarding strategies, the performance is slightly better when only personal data are used. The favouring of personal data allows the creation of a network more robust against a clique created by selfish nodes, i.e., prevents selfish nodes from obtaining additional advantages by being a member of a clique.

References

1) Du, X.: Backbone quality-of-service routing protocol for heterogeneous mobile ad hoc networks, *Advances in Wireless Ad Hoc and Sensor Networks*, ch.1, pp.1–34, Springer (2010).
2) Jensen, C.D. and Connell, P.O.: Trust-based route selection in dynamic source routing, *Proc. 4th International Conference on Trust Management* (*iTrust 2006*), LNCS, Vol.3986, pp.150–163, Springer (2006).
3) Corson, S. and Macker, J.: Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations, IETF RFC 2501 (1999). (Online) Available from http://www.ietf.org/rfc/rfc2501.txt
4) Buttyan, L. and Hubaux, J.P.: Nuglets: A virtual currency to stimulate cooperation in self-organized mobile ad hoc networks, Swiss Federal Institute of Technology, Technical Report DSC/2001/001 (2001).
5) Marti, S., Giuli, T., Lai, K. and Baker, M.: Mitigating routing misbehavior in mobile ad hoc networks, *Proc. ACM/IEEE 6th International Conference on Mobile Computing and Networking* (*MobiCom 2000*), pp.255–265 (2000).
6) Michiardi, P. and Molva, R.: Simulation-based analysis of security exposures in mobile ad hoc networks, *Proc. European Wireless Conference* (2002).
7) Kwok, Y.-K.: Incentive issues in peer-to-peer systems, *The Handbook of Computer Networks*, Bidgoli, H. (Ed.), John Wiley and Sons, Vol.3, ch.146, pp.168–188 (2007).
8) Carruthers, R. and Nikolaidis, I.: Certain limitations of reputation-based schemes in mobile environments, *Proc. 8th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems* (*MSWiM 2005*), pp.2–11 (2005).
9) Buchegger, S. and Boudec, J.-Y.L.: Performance analysis of the confidant protocol, *Proc. 3rd International Symposium on Mobile Ad Hoc Networking and Computing* (*MobiHoc 2002*), pp.226–236 (2002).
10) Michiardi, P. and Molva, R.: Core: A collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks, *Proc. 6th Conference on Security Communications, and Multimedia* (*CMS 2002*), pp.107–121 (2002).
11) Buchegger, S. and Boudec, J.-Y.L.: The effect of rumor spreading in reputation systems for mobile ad-hoc networks, *Proc. Workshop on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks* (*WiOpt 2003*), pp.131–140 (2003).
12) He, Q., Dapeng, W. and Khosla, P.: SORI: A Secure and Objective Reputation-Based Incentive Scheme for Ad-hoc Networks, *Proc. Wireless Communications and Networking Conference* (*WCNC 2004*), Vol.2, pp.825–830 (2004).
13) Buchegger, S. and Boudec, J.-Y.L.: Self-policing mobile ad hoc networks by reputation systems, *IEEE Communications Magazine, Special Topic on Advances in Self-Organizing Networks*, Vol.43, No.7, pp.101–107 (2005).

14) Yan, L. and Hailes, S.: Cooperative packet relaying model for wireless ad hoc networks, *Proc. 1st ACM International Workshop on Foundations of Wireless Ad Hoc and Sensor Networking and Computing*, pp.93–100, ACM (2008).
15) Milan, F., Jaramillo, J. and Srikant, R.: Achieving cooperation in multihop wireless networks of selfish nodes, *Proc. Workshop on Game Theory for Communications and Networks*, ACM (2006).
16) Yan, L. and Hailes, S.: Designing incentive packet relaying strategies for wireless ad hoc networks with game theory, *Wireless Sensor and Actor Networks II*, pp.137–148, Springer, Boston (2008).
17) Seredynski, M., Ignac, T. and Bouvry, P.: Probabilistic packet relaying in wireless mobile ad hoc networks, *Proc. 8th International Conference on Parallel Processing and Applied Mathematics* (*PPAM 2009*), Vol.6067, LNCS, pp.31–40 (2010).
18) Jøsang, A., Ismail, R. and Boyd, C.: A survey of trust and reputation systems for online service provision, *Decision Support Systems*, Vol.43, No.2, pp.618–644 (2007).
19) Rasmusson, L. and Jansson, S.: Simulated social control for secure internet commerce, *Proc. 1996 Workshop on New Security Paradigms* (1996).
20) Hubaux, J.-P., et al.: Cooperation in wireless networks. http://winet-coop.epfl.ch/ (last checked Oct. 2010).
21) Giordano, S. and Urpi, A.: Self-organized and cooperative ad hoc networking, *Mobile Ad Hoc Networking*, Basagni, S., Conti, M., Giordano, S. and Stojmenovic, I. (Eds.), ch.13, pp.355–371, Wiley-IEEE Press (2004).
22) Seredynski, M. and Bouvry, P.: Direct vs. indirect reciprocity trust system in ad hoc networks (short paper), *Proc. 4th IFIP WG 11.11 International Conference on Trust Management* (*IFIPTM 2010*), Morioka, Japan, pp.111–118 (2010).
23) Hu, J. and Burmester, M.: Cooperation in mobile ad hoc networks, *Guide to Wireless Ad Hoc Networks*, ch.3, pp.43–57, Springer (2009).
24) Mundinger, J. and Boudec, J.-Y.L.: Analysis of a reputation system for mobile ad-hoc networks with liars, *Performance Evaluation*, Vol.65, No.3-4, pp.212–226 (2008).
25) Buchegger, S. and Boudec, J.-Y.L.: A robust reputation system for peer-to-peer and mobile ad-hoc networks, *Proc. 2nd Workshop on the Economics of Peer-to-Peer Systems* (2004).
26) Seredynski, M. and Bouvry, P.: The cost of altruistic punishment in indirect reciprocity-based cooperation in mobile ad hoc networks, *Proc. 6th IEEE/IFIP International Symposium on Trusted Computing and Communications* (*TrustCom 2010*), Hong Kong, China, pp.749–755 (2010).
27) Seredynski, M. and Bouvry, P.: Evolutionary game theoretical analysis of reputation-based packet forwarding in civilian mobile ad hoc networks, *Proc. 22nd IEEE International Parallel and Distributed Processing Symposium, NIDISC Workshop* (May 2009).
28) Jubin, J. and Turnow, J.D.: The DARPA packet radio network protocols, *Proc. IEEE*, Vol.75, No.1, pp.21–32 (1987).
29) Feldman, M., Papadimitriou, C., Chuang, J. and Stoica, I.: Free-riding and whitewashing in peer-to-peer systems, *IEEE Journal on Selected Areas in Communications*, Vol.24, No.5, pp.1010–1019 (2006).

(Received November 1, 2010)
(Accepted April 8, 2011)
(Released July 6, 2011)

Marcin Seredynski was born in 1979. In 2004 he received his M.Sc. from Faculty of Electronics and Information Technology of the Warsaw University of Technology. In 2009 he defended his Ph.D. thesis entitled An Evolutionary Approach towards Cooperation Enforcement in Ad Hoc Networks with the University of Luxembourg and Polish Academy of Sciences. Currently he is a scientific collaborator at the Interdisciplinary Centre for Security, Reliability and Trust in Luxembourg. His research interests include evolutionary game theory, trust management and mobile ad hoc networks.

Pascal Bouvry earned his Ph.D. degree (1994) in computer science at the University of Grenoble, France. He is now professor at the Faculty of Sciences, Technology and Communication of the University of Luxembourg and heading the Computer Science and Communication research unit. Pascal Bouvry is specialised in parallel and evolutionary computing. His current interest concerns the application of Nature-inspired computing for solving reliability, security, and energy-efficiency problems.

Dominic Dunlop was born in 1952. He received his B.Eng. and M.Sc. from Bradford and Luxembourg Universities in 1975 and 2009 respectively. Starting his career as a broadcast engineer with the BBC, he moved into computers, being involved in microprocessor support, early commercial UNIX systems, computer-aided lexicography, and banking. Following master's studies related to high-performance computing, he is currently engaged in systems support at the University of Luxembourg. He is a member of IEEE.