Privacy-Preserving Interest-Casting in Opportunistic Networks

Gianpiero Costantino, Fabio Martinelli, Paolo Santi
IIT-CNR, Pisa, Italy
Email: name.surname@iit.cnr.it

***Abstract*—Message forwarding is a fundamental brick to spread information among users in opportunistic networks. In this paper, we consider the recently proposed interest-casting networking primitive for opportunistic networks, according to which a packet generated by a sender should be delivered to all users in the network – potentially unknown to the sender – sharing similar interests. However, the current implementation of interest-casting assume users exchange their interest profiles to take forwarding decisions, thus revealing very sensitive information to strangers. In this work, we approach for the first time the problem of designing an interest-casting protocol while not revealing sensitive information during the forwarding and message delivery process. In particular, we present a privacy-preserving mechanism based on the well-known Millionaires' problem allowing users to discover whether they have similar interests without disclosing private information. Based on this mechanism, we propose four different privacy-preserving forwarding protocols to realise interest-casting in opportunistic networks, and we compare their performance on a real-world mobility trace.**

***Index Terms*—Wireless Networks, The Millionaire's Problem, Simulations, Security Attacks, Simulations**

I. Introduction

Nowadays, people are surrounded by portable devices, such as smartphone, PDAs, and laptops. Besides allowing direct connection to the Internet through, e.g., a 3G connection, these devices allow users to create direct, device-to-device wireless links exploiting Bluetooth or WiFi connections. Thus, networks can be formed based on these direct communication links which, however, typically display a very sparse and mostly disconnected network topology. Opportunistic Networks *(OppNets)* considered in this paper are networks composed of sparsely deployed portable devices, where direct communication opportunities between users are exploited to spread information within the network.

A recently proposed technique used to optimise spreading of information within OppNets exploits the notion of user interests [1]. In fact, people can be interested in receiving information regarding a particular topic, e.g., Book or Music. Suppose for instance user *Alice* has a high interest in books, and she is then interested in getting as many news as possible about this topic. After setting this information in her smartphone in form, e.g., of interest profile, she turns the Bluetooth/WiFi interface on and starts walking. When the device detects another user, it starts querying it about the topic Book. Let *Bob* be the owner of the new device discovered; if he is also interested in the same topic, they may share their knowledge.

The technique mentioned above proposes an easy way of sharing messages among users interested in same topics. However, it is implicitly based on a fully trusted network model, according to which each user in the network, even if stranger, can be completely trusted. Experience taught us that unfortunately this assumption does not hold in real world, where malicious user behaviour emerges even in relatively small size network. Clearly, the above mentioned approach is doomed to perform poorly in an environment populated by malicious users, whom could easily gain sensitive information about other users' interests, and exploit this information to clone identities, disrupt information propagation, and so on.

In this work, we approach for the first time the problem of realising an interest-cast primitive, according to which a message is delivered to all users sharing the same interest of the sender, in a privacy-preserving manner, i.e., disclosing only minimal information about user interests. The key mechanism proposed in this paper is a generalisation of the well-known protocol for solving the Millionaire's problem, which allows computing whether user interests about a particular topic are similar enough without disclosing their private interest value. Based on this key mechanism, we introduce four different ways of forwarding messages among users that are interested in same topics, and we evaluate their performance using a real-world mobility trace.

This paper is structured as follows. Section II recalls existing forwarding models within OppNets and solutions to defend users against well-know attacks. Section III clarifies the concept of interest-casting. In Section IV, we formalise our forwarding protocols. Section V introduces an optimised version of The Millionaire's Problem to use within OppNets. In Section VI, findings of our simulations are shown. Finally, Section VII concludes the paper.

II. Related Work

Routing based on similar interests has already tackled within OppNets [1], [2], [3]. However, security problems may raise from users' interactions, for example a user may disclose private information that can be used by an attacker. So far, solutions provided by the researchers community do not consider technique to preserve the privacy of participants within OppNets.

A reputation system for OppNets is presented in [4]. Through this system malicious users are discovered and excluded as forwarders in further communications. Their system consists in evaluating the behaviour of participants using both direct and indirect observation. So, each time that a packet is correctly forwarded, a receiver sends a particular message called Positive Feedback Message *(PFM)* to the sender of the communication. In this way, the sender increases the reputation linked to that forwarder, and also it knows that the packet was correctly forwarded.

Trust relationship is built using friend ties in [5]. The reputation system is based on the number of common friends that a user has with others. In particular, when two participants of an OppNet meet each other, they exchange their friends' list and so they build a graph of friend connections. The closer is a friend to the root, the higher is his/her trusting level. The authors assert that they are able to limit the maximum number of multiple identities that can be generated with a *Sybil attack*.

In [6], the authors introduce a technique to highlight trusted devices through direct and indirect observation. They use an ontology to create policies in which participants have to agree in order to obtain a service. Similar to existing trusting systems, direct observation are obtained by direct past experience while indirect ones are got using recommendations. In addition, the authors consider a fading parameter that decreases the reputation values of users that are not still collaborating.

An alternative to push users to collaborate is proposed in [3]. The authors try to force users to be cooperative by carrying out messages that they consider important and also useful for others. In this scenario the authors consider two type of messages: primary and secondary. A message that is important for a device itself is considered as primary, otherwise it is secondary. However, the latter type of message can be very useful for others, and so carrying secondary messages proves cooperation of devices. Finally, by means of barter a user is able to get a message only if it provides the same number of messages to the other participant.

Although part of these works use security approaches to reduce the negative impact of malicious users within OppNets, our main goal is to ensure that trust links among participants cannot unveil users' private information. In fact, users friendship is based on similar interest and a routing system built exploiting this technique can be easily attacked by dishonest users when private information are disclosed. In addition, to the best of our knowledge secure multiparty techniques, such us that one implemented in *The Millionarire's Problem*, have not been considered in this field from researchers with the aim to minimise the risk that a malicious user may run an attack using sensible data got by past interactions.

III. Network model and preliminaries

We consider an OppNet composed of n nodes (users), and denote the set of nodes in the network by $\mathcal{N}$. Similarly to [1], we assume user interests can be modelled as an m-dimensional vector in a common m-dimensional *interest space*, where $m \ll n$. More formally, the *interest profile* of user A is defined as:

$$I_A = (a_1, \ldots, a_m) \ ,$$

where $a_i \in [0, max]$ is an integer representing A's interest in the i-th topic of the interest space. Note that interests are expressed as integers in the range $[0, max]$, with 0 representing no interest and max (an arbitrary integer > 0) representing maximum interest[1]. Although our approach can be extended to deal with the case of two users with the same interest profile, to simplify presentation in the following we make the assumption that no two users in the network have the same interest profile.

In this paper, we are concerned with realizing a privacy preserving *interest-casting* primitive, where the interest-casting primitive is defined as follows [1]. Let S be a user denoted as the message *source*. The message M generated by S must be delivered to all nodes in the set $\mathcal{D}(S,\gamma)$, where

$$\mathcal{D}(S,\gamma) = \{U \in \mathcal{N} | sim(U,S) \geq \gamma\} \ ,$$

where $sim(U,S)$ is a similarity metric used to express similarity between U and S's interest profiles, with relatively higher similarity values representing relatively more similar interests, and γ is the *relevance threshold*. Set $\mathcal{D}(S,\gamma)$ is called the set of *relevant destinations*, and in principle it is not known in advance to node S. Instead, set $\mathcal{D}(S,\gamma)$ is implicitly defined by S's interest profile, and by the relevance threshold γ. Furthermore, users in set $\mathcal{D}(S,\gamma)$ are not assumed to undertake any explicit action (e.g., subscribing to a thematic channel) to be able to receive message M. This is in sharp contrast to more traditional networking primitives such as multicast, where the set of destinations is known in advance to the source, and publish/subscribe, where subscriptions to thematic channels are mandatory.

More specifically, in this paper we define the following similarity metric between interest profiles, which we call *vector-component-wise* (vcw) similarity metric. Let $S = (s_1, \ldots, s_m)$ and $U = (u_1, \ldots, u_m)$ be the interest profiles of users S and U, respectively. We have:

$$vcw(U,S,\lambda) = \left\{ \begin{array}{ll} 1 & \text{if } \forall i \in \{1,\ldots,m\},\ |u_i - s_i| \leq \lambda \\ 0 & \text{otherwise} \end{array} \right. ,$$

where $\lambda \in [0, max]$ is an integer parameter used to narrow/widen the scope of the interest-cast. More specifically, by setting $\gamma = 1$, we have that $\mathcal{D}(S,1) = \mathcal{N}$ if $\lambda = max$, and $\mathcal{D}(S,1) = \{S\}$ if $\lambda = 0$. To simplify notation, in the following we denote $\mathcal{D}(S,1)$ by $\mathcal{D}(S)$.

We assume message M generated by S is characterized by a TimeToLive (TTL), i.e., a time interval beyond which the information contained in the message is considered no longer valuable. The goal of the forwarding protocols described in the following is delivering a copy of M to as many nodes in $\mathcal{D}(S)$ as possible within time TTL since its generation at S. More specifically, for a given forwarding protocol $\mathbf{F}$, and denoting by $\mathbb{P}_{\mathbf{F}}(U)$ the property "user U received a copy of

[1] The notion of interest profile can be straightforwardly extended to represent also information about a user's habits, such as living in a certain neighborhood, working in a certain place, and so on. For details, see [1].

M within time TTL under forwarding scheme $\mathbf{F}$", we define the set of *covered nodes* $\mathcal{C}(\mathbf{F})$ as follows:

$$\mathcal{C}(\mathbf{F}) = \{U \in \mathcal{N} | \mathbb{P}_{\mathbf{F}}(U) \text{ is } \mathbf{true}\} \ .$$

We can now define the following *precision* and *coverage* metric (equivalent to the precision and recall metrics well known in information retrieval [7]). We have:

$$Prec(\mathbf{F}) = \frac{|\mathcal{C}(\mathbf{F}) \cap \mathcal{D}(S)|}{|\mathcal{C}(\mathbf{F})|}$$

and

$$Cov(\mathbf{F}) = \frac{|\mathcal{C}(\mathbf{F}) \cap \mathcal{D}(S)|}{|\mathcal{D}(S)|} \ ,$$

where $Prec(\mathbf{F}) = 1$ represents maximum possible precision (M is delivered only to nodes in $\mathcal{D}(S)$), and $Cov(\mathbf{F}) = 1$ represents maximum possible coverage (M is delivered to all nodes in $\mathcal{D}(S)$). Ideally, we would like to design a forwarding protocol simultaneously achieving maximum precision and coverage. However, as we shall see in the following, the two metrics above are often in contrast with each other, and the most adequate tradeoff between them should be sought.

IV. Forwarding protocols

In the following, we will present privacy-preserving versions of the following forwarding protocols

- *direct delivery* ($\mathbf{DD}$): strictly speaking, this is not a forwarding protocol: source node S delivers a copy of M whenever it has a communication opportunity with a node $U \in \mathcal{D}(S)$. Message forwarding is not allowed: only S can deliver copies of M to relevant destinations.
- *2-hop forwarding* ($\mathbf{2H}$): similarly to $\mathbf{DD}$, node S delivers a copy of M to each node in $\mathcal{D}(S)$ it gets in touch with. However, in this case forwarding of a copy of M to other nodes is allowed. More specifically, any node U in $\mathcal{D}(S)$ holding a copy of M can deliver a copy of it to any other node V it meets under the condition that $vcw(U,V) = 1$. Note that, in order to preserve a minimum level of precision, forwarding can occur only along paths composed of two hops at most: in particular, any node which receives a copy of M from a node $U \neq S$ (as node V above) is not allowed to further forward the message.
- *restricted 2-hop forwarding* ($\mathbf{R2}$): this protocol is similar to protocol $\mathbf{2H}$, with the only difference that message forwarding and delivery to destination is driven by the condition that $vcw_{\lambda'}(U,S) = 1$, where $\lambda' < \lambda$, and $vcw_{\lambda'}()$ is the vcw similarity metric computed using λ', instead of λ, to define the component-wise similarity threshold. As we shall see in the following, this restrictive choice concerning forwarding allows, by suitably tuning parameter λ', to optimally address the precision/coverage tradeoff.

Protocol $\mathbf{DD}$ has maximum precision, since only nodes in set $\mathcal{D}(S)$ can receive a copy of M. However, this protocol likely displays low coverage, since no forwarding mechanism is realised; i.e., relatively few communication opportunities can be exploited to deliver M to relevant destinations. On the other hand, protocol $\mathbf{2H}$ aims at increasing coverage introducing two-hops forwarding. However, this comes at the price of precision: in fact, it is easy to see that under protocol $\mathbf{2H}$ also nodes in $\mathcal{N} - \mathcal{D}(S)$ can receive M. Finally, protocol $\mathbf{R2}$ aims at achieving an optimal tradeoff between precision and coverage by tuning parameter λ'. In particular, it can be shown (the formal proof is omitted due to lack of space) that, by setting $\lambda' = \lambda/2$, protocol $\mathbf{R2}$ ensures maximum precision of 1. In the following, we call this version of $\mathbf{R2}$ protocol $\mathbf{E2}$, to emphasize the fact that under this protocols the message is delivered only to nodes in $\mathcal{D}(S)$.

V. Our proposal

In the following, we assume that a users interest is defined in a fixed range —from 1 to 100—, reflecting the fact that a participant can be interested or not in receiving messages about a topic. Table I reports a possible *Alice's interest profile*, with a *degree of interest* expressed for each topic. Similarly, Bob's interest profile is shown in Table II. When Bob meets Alice, he would be able to share his own messages with Alice and vice-versa. In particular, for sharing we consider both download and upload of messages from a user to another one. The action of scanning an interest profile of another users can easily show the similarity in interests as researchers in [1] present. However, this technique introduces several problems concerning user's privacy. In fact, if Bob were a malicious user he would get Alice's private information from her interest profile.

TABLE I
Alice's interest profile

Cinema	Book	Music	...	Car
40	30	60	...	10

TABLE II
Bob's interest profile

Cinema	Book	Music	...	Car
30	60	70	...	80

Following we present some attacks that Bob may perform:

- He may discover the amount of Alice's interest for each topic.
- He may download Alice's messages by introducing himself as interested in the same Alice's topics.
- He may reveal information obtained by the Alice's interest profile to another user *(Collusion Attack)*.

Hence our opinion is that a users' interest profiles must be kept private, and no information must be disclosed by a user when profiles are matched.

Nowadays researchers have provided a lot of definitions regarding trust [6], [8], [9]. In this context we say that: *Alice can trust in Bob whether for selected topics, they discover to have a similar interest without revealing any private information.* In fact, when Alice meets Bob, she challenges him only on a sub-set of topics. In particular, this sub-set is a *vector of topics* and its dimension, denoted as $|v_t|$, is not fixed.

To help Alice and Bob establishing whether they can trust each other, we adopt a solution proposed in the cryptographic field introduced by Yao [10] known as "The Millionaire's Problem". It will be used to compare, in a privacy-preserving fashion, the interest of users.

A. An efficient solution of The Millionaire's Problem

Goal of the *"The Millionaire's Problem"* is to compare two numbers, "i" and "j", and to discover if:

$$i \leq j \qquad \text{or} \qquad i > j \tag{1}$$

However, this comparison must not leak out any information about the two numbers. if Alice holds "i" and Bob has "j", by running *"The Millionaire's Problem"*, they would get to know which number is higher without giving any information regarding the number kept. So, in the 1982 Yao presented his solution that belongs to the secure multiparty computation field. He assumes that users must complete all steps of the protocol. In fact, since Alice will find out the result of the comparison in 1, she may decide to not inform Bob about the result. In this way, Alice knows which number is higher, while Bob does not have the result about the comparison. As explained in [11], most researchers assume that users who participate in a computation are semi-honest. In particular, a semi-honest participant is a user who properly follows the protocol, but he/she is able to record all information derived throughout the protocol's steps.

The author in [12] report that the computational complexity of the Yao's solution required to compare *i,j* is exponential in n. According to this, we need two main requirements from *"The Millionaire's Protocol"* in order to perform our goal, these are:

- it must be efficient to be easily executed by mobile devices.
- it must not disclose any secret information to users when they run the protocol.

In [13], [14], two more recent and efficient works about *"The Millionaire's Problem"* are proposed. In particular, these solutions works with asymmetric cryptography, e.g. *RSA*. In addition, in the paper [13] is also proposed a version that uses symmetric keys and real numbers. Nevertheless, computational results, obtained using an old Pentium III/450Mhz, prove that the hardware of recent mobile devices is able to run these solutions of *"The Millionaire's Problem"*.

B. The Millionaire's Problem applied in OppNets

In Section V, we gave our definition of trust considering two users belonging to an OppNet who would share messages for a given topic. For istance, Alice and Bob have one number each: i is hold by Alice and j by Bob. Now, by running the protocol for the *"The Millionaire's Problem"*, they discover whether their interests are similar, where similar means that the interest degrees in the specific topic differ for at most λ. Hence, Alice must verify that:

$$|i - j| \leq \lambda \; . \tag{2}$$

After verifying the condition 2, Alice shares her messages with Bob and vice-versa. However, that condition cannot be performed with only one run of *"The Millionaire's Problem"*. So, Alice has to execute the protocol in Fig. 1 — where ϵ is an arbitrarily small positive number.

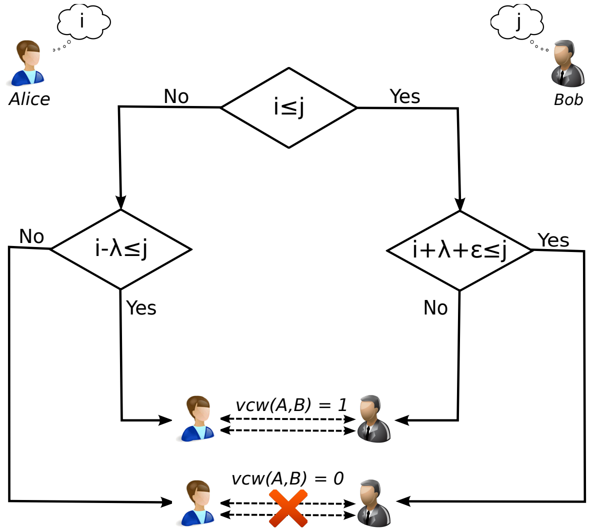

Fig. 1. Protocol flow to discover similarity in an interest

By having two numbers —one for each participant—, a user run *"The Millionaire's Problem"* to compare them. However, the flow depicted in Fig. 1 is used to compare Alice's and Bob's degree of interest for a single topic of interest. To do it for the entire profile or part of it, the protocol in Fig. 1 is used. Hence, Alice selects a subset of topics from her interest profile, and she challenges Bob using only those topics. By doing as shown in Fig.2, she is able to speed up her challenge with Bob, at the price of loosening the trust relationship with Bob.

At the end, Alice knows which topics they will share, but neither Alice nor Bob are able to know the exact degree of interest the other party has in those topics.

C. Security analysis

Using the aforementioned technique, we showed that a single user is not able to get sensible information of another participant. On the other side, by running the protocol without using the Millionaire's Problem, a malicious user (Bob) may run the *Collusion Attack* with another bad person *(Peter)*. As far as we know this attack is not considered in OppNets literature, and we want to analyse whether our approach is sound against the Collusion Attack. We suppose that there are three main actors: Alice, Bob and Peter. In particular, Bob and Peter are colluding and Bob starts an interaction with Alice in order to get her friendship. At this point two cases can appear:

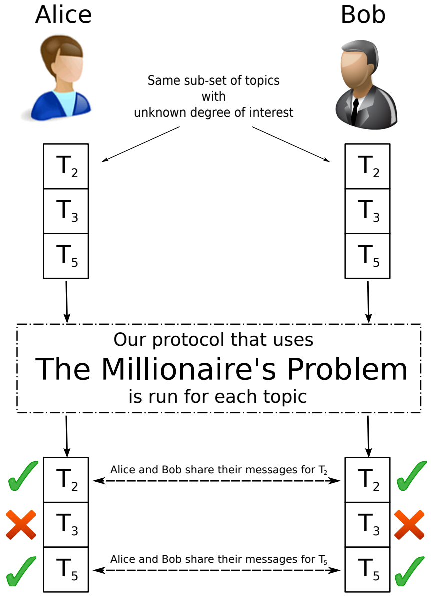

Fig. 2. Protocol flow to discover similarity in an interest

Case 1: Alice challenges Bob on a subset of topics and she recognises that their interests are not similar. Bob does not get any information from this interaction and as consequence of it, he is not able to "sell" any information to Peter because Alice did not disclose her interest profile. In this scenario our solution is robust against the *Collusion Attack*.

Case 2: If Bob is a friend of Alice for a particular interest, like books, he may reveal his knowledge to Peter. So, he may meet Alice asking for becoming her friend. Although Peter obtains j from Bob for a topic and he reuses it with Alice, she, however, can decide to challenge Peter using a different subset of topics. Hence, Peter may not have any chance to exploit his attack. When Alice starts to interact with Peter, she selects one or more topics from her vector. Since Peter has at least one Alice's topic, he hopes that she will choose at least one same topic again. In such a case, Peter is actually able to get some advantage from his collusion with Bob.

VI. Simulations

Compared to the traditional techniques of delivering data using a fixed destination specified by an IP address, in our approach messages are sent only to those devices that share a same interest about a particular topic. In this section, we evaluate the performance of the four message forwarding protocols introduced in Section IV. To analyse the results regarding our propagation models we built a simulator, written in JAVA. In particular, the simulator was developed to work with the database given by the Massachusetts Institute of Technology *(MIT)* that provides the users mobility pattern. In fact, MIT-researchers collected mobility traces of 97 users from July 2004 to April 2005[2]. The information available show different aspects about the users, such as communication, devices in proximity, location, and activity information. However, in the dataset is not included any information regarding users interest profile. Thus, to fill the gap, we decided to run a survey within our research centre *(National Research Council)* in which we gave to 97 users an online and anonymous survey asking them to fill with his/her degree of interest each topic. This value was ranged from "1" to "100" and it was required for 15 topics. Then, we associated in a random way, each survey to a user of the MIT database.

Briefly, our simulator works in the following way: it scans all users who meet each other, i.e. their Bluetooth interface discovers in proximity[3] other Bluetooth devices. We assume that each generated in the network is assigned with a single, randomly chosen topic of the interest profile. Furthermore, TTL is assumed to be infinite in our simulations, and the similarity threshold λ is set to 10. When a user *"A"* meets a user "B" in her neighbourhood, for each packet that *"A"* has in her queue, she checks the similarity of interest for that packet-Topic with *"B"*. If the condition (2) is verified, then *"A"* sends that message to *"B"*. Subsequently, depending on the forwarding protocol, also *"B"* may forward the packet to other users.

We ran several simulations and evaluated the following performance metrics:

- *Coverage*, i.e., the fraction of nodes in set $\mathcal{D}(S)$ that received the message.
- *Precision*, i.e., the ratio between the number of devices in $\mathcal{D}(S)$ that received the message, and the total number of nodes in the network that received the message.
- *Delay*, i.e., the average delay with which the message is received at intended destinations;

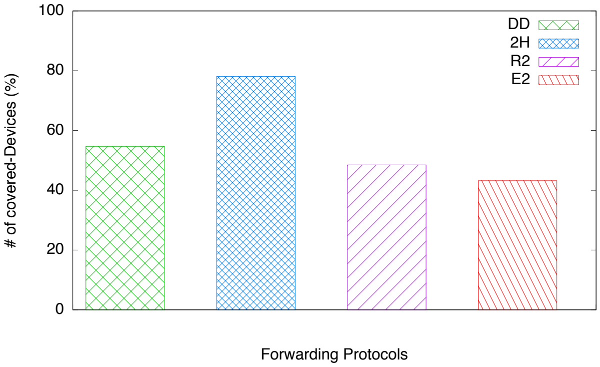

Fig. 3. Devices covered in average

Fig.3 shows the coverage provided by each protocol. In particular, we consider a single packet sent by a source device. Results obtained are the average of 10 simulations ran for each protocol, and the top of the chart expresses the cardinality of set $\mathcal{D}(S)$. As expected, the best coverage is provided by the *2H*-protocol, which is a multi-hop protocol with loose forwarding rules. However, the latter situation

[2] Simulations were run using a time-window from July to December 2004.

[3] Bluetooth devices within 5-10 meters.

makes it possible to deliver that packet to devices that should not have it – see Table III reporting the protocols precision. A good coverage is also provided by the *DD*-protocol. It is single-hop and cannot send message to devices outside set $\mathcal{D}(S)$. Finally, *R2*[4] and *E2*[5]-protocol have a lower forwarding threshold and this makes difficult to reach a high number of devices. However, the results obtained are positive.

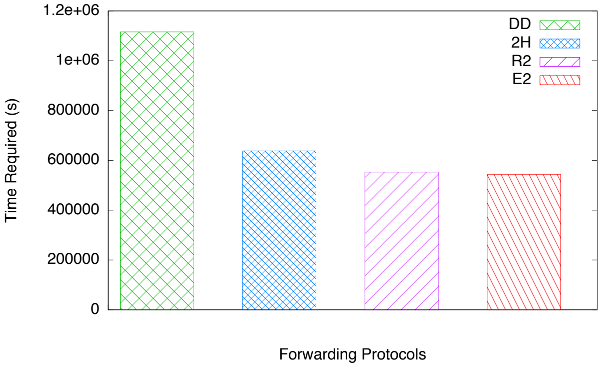

Fig. 4. propagation delay in average

Fig.4 shows the average delay experienced with the four forwarding protocols. In this case, the *R2* and *E2*-protocol result the best ones. On the other hand, although *DD*-protocol has a good coverage, its propagation delay is very high. In fact, it is a single-hop protocol and this does not make easier to reach all devices in a short time.

TABLE III
PRECISION IN AVERAGE

DD	2H	R2	E2
100%	60.7%	99.7%	100%

Table III reports the *precision* obtained for each protocol in average of 10 simulations. As expected, *DD* and *E2*-protocol have the maximum precision since they are designed to avoid to share packets with devices that have a different degree of interest. Instead, *2H* and *R2* protocol can deliver a message to a device outside set $\mathcal{D}(S)$. In particular, the less accurate protocol results *2H* since a node that receive a packet can resend it to others applying the same condition that the sender applied to it. As seen from Table III, the precision of *2H*-protocol is not so high, demonstrating that the protocol fails short in accuracy. The other version the same protocol, i.e. *R2* protocol, is very accurate, in fact its 0.3% to reach the top is negligible.

Overall, we obtain that *2H*-protocol has the best tradeoff considering coverage and propagation delay. However, it is penalised by less accuracy in correctly delivering packets. On the other hand, the single-hop *DD*-protocol is very accurate with a good coverage, but it displays a very slow propagation time. Finally, *R2* protocol is more performant than *E2*-protocol due to its higher coverage.

VII. Conclusion - Future Works

In opportunistic networks users, can share messages when they have a similar interest for a particular topic. According this issue, we introduced different techniques to forward messages in order to obtain high coverage and accuracy of users interested in that information. Moreover, to defend users by malicious ones in getting private information, we decided to adopt an optimised version of the Millionaire's Problem. In fact, by using that cryptographic-technique, two users are able to discover if they can trust each other since their interest is similar without disclosing sensitive information.

Finally, to study the performance of the different forwarding protocols, we implemented a simulator exploiting real users mobility logs. Proposed protocols have been analysed using different parameters, such as users coverage, message-delay propagation and accuracy in propagation. In particular, the results show that our forwarding protocols obtains a good level of coverage while keeping the accuracy of the receivers very high.

In this version of the simulator we did not consider users who took malicious behaviours. In fact, simulations showed results in which messages are propagated only when a receiver is interested in getting messages. Hence, we did not show how malicious users can modify the propagation rate of our forwarding protocols. Thus, we keep this goal as future works.

References

[1] A. Mei, G. Morabito, P. Santi, and J. Stefa, "Social-aware stateless forwarding in pocket switched networks," in *IEEE Infocom*, 2011.

[2] X. X. Ting Ning, Zhipeng Yang and H. Wu, "Incentive-aware data dissemination in delay-tolerant mobile networks," 2011.

[3] L. Buttyán, L. Dóra, M. Félegyházi, and I. Vajda, "Barter trade improves message delivery in opportunistic networks," *Ad Hoc Networks*, vol. 8, no. 1, pp. 1 – 14, 2010.

[4] N. Li and S. K. Das, "A trust-based framework for data forwarding in opportunistic networks," *Ad Hoc Networks*, vol. In Press, 2011.

[5] S. Trifunovic, F. Legendre, and C. Anastasiades, "Social trust in opportunistic networks," in *INFOCOM IEEE CCCW*. IEEE, Mar. 2010, pp. 1 – 6.

[6] M. R. P. Gonçalves, E. D. S. Moreira, and L. A. F. Martimiano, "Trust and privacy: Informal ways to assess risk on opportunistic exchanges," *Inter. Journal of Computer Science Applications*, vol. 6, pp. 66 – 85, 2009.

[7] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*. New York: ACM Press - Addison-Wesley, 1999.

[8] S. Marsh, "Formalising trust as a computational concept," PhD thesis, University of Stirling, UK, 1994.

[9] A. Jøsang, R. Ismail, and C. Boyd, "A survey of trust and reputation systems for online service provision," *Decision Support Systems*, vol. 43, no. 2, pp. 618 – 644, March 2007.

[10] C. Andrew and C. Yao, "Protocols for secure computations," in *23rd IEEE Symposium on FOCS*, 1982, pp. 160 –164.

[11] O. Goldreich, *Foundations of Cryptography*. Cambridge University Press, 2001, vol. Basic Tools.

[12] M. J. Freedman, K. Nissim, and B. Pinkas, "Efficient Private Matching and Set Intersection," in *EUROCRYPT*, ser. LNCS, vol. 3027, 2004, pp. 1 – 19.

[13] S. Li, D. Wang, and Y. Dai, "Symmetric cryptographic protocols for extended millionaires' problem," *Information Sciences*, vol. 52, no. 6, pp. 974 – 982, 2009.

[14] I. Ioannidis and A. Grama, "An efficient protocol for Yao's millionaires' problem," *System Sciences, 2003. Proc. of the 36th Annual Hawaii Int. Conf.*, p. 6, 2003.

[4] Packets forwarded using the *R2*-protocol have λ set to 8.

[5] Packets forwarded using the *E2*-protocol have λ set to 5.