Content Dissemination on Location-based Communities: a Comparative Analysis

Elena Pagani
Computer Science Dept., Università degli Studi di Milano
CNR-IIT, National Research Council, Italy
Email: pagani@di.unimi.it

Gian Paolo Rossi
Computer Science Dept.
Università degli Studi di Milano, Italy
Email: rossi@di.unimi.it

***Abstract*—This paper focuses on content dissemination in location-centered communities and provides the first comparative analysis of two forwarding algorithms on real scenario, namely, ProfileCast – which has been on purposely designed for this environment – and InterestCast – which by contrast addresses more general settings. The paper provides quantitative evaluation of relevant metrics (i.e. community coverage, delivery delay, energy/message efficiency) to be considered whenever attempting to spread contents to the persons that are used to visit the same location. Moreover, the experiment allows to give an insight on the problems arising when deploying these protocols on real settings and an empirical evaluation of two different approaches. ProfileCast leverages mechanisms to automatically extract the intrinsic characteristics of the users from their behavior pattern; a content generated by a node is implicitly addressed to users with similar behavior as the source. InterestCast matches content tags against interests explicitly expressed by the users.**

I. Introduction

In the era of mobile and pervasive computing, opportunistic networks (ONs) leverage human encounters to provide an intermittently-connected network where the diffusion of delay tolerant information through the personal devices of individuals mimics the human communication on the grapevine. The current growing interest in human social interactions inspires the creation of novel communication paradigms for ONs, hastening the shift from unicast interactions of early ONs to new form of anycast communications. In fact, one-to-many interactions are considered as more suitable to support emerging services that exploit the human attitude to organize in communities where common behavior and interests are shared. The content dissemination inside a community advocates a departure from the classical IP-addressing style to privilege new policies in which the binding content-recipients is not provided by the sender, but directly executed by specific recipients with an interest in it. Human encounters drive the information flow towards potential recipients that extract it from the stream when content type and personal interest match.

In line with the described trend, all recent proposals are addressing the objective of casting contents on different typologies of communities. ProfileCast [9] considers location-centered communities pooling individuals sharing the attitude to visit certain places. SocialCast [3] considers communities deriving from social ties and assumes that users with social relationship have the attitude to meet one another more often than with other users. ContentPlace [2] assumes that users belong to social communities and that communities are bound to physical places. InterestCast (ICast [11]) is able to chase users interests decoupling content tags from locations and social communities and, thus, should be able to operate on either types (location, social or other) of communities.

This paper focuses on content dissemination in location-centered communities and provides the first comparative analysis of two forwarding algorithms on real scenario, namely, ProfileCast – which has been on purposely designed for this environment – and InterestCast – which by contrast addresses more general settings. The paper provides quantitative evaluation of relevant metrics (i.e. community coverage, delivery delay, energy/message efficiency) to be considered whenever attempting to spread contents to the persons that are used to visit the same location.

This analysis gives an insight on the problems arising when deploying these protocols on real settings and an experimental evaluation of two different approaches. ProfileCast interestingly tries to compare human behavior by working on similarities between two encounters, while InterestCast requires to explicitly name the target interest. Both have a wide range of application although the former has a higher potential to flexibly adapt to a given target behavior enabling, at the same time, the identification of the variety of behavior shades around the target one. However, our analysis shows that the second outperforms the first because, until now, it is still very difficult to derive the correct parameters to apply when comparing similarities.

II. System Assumptions and Problem Definition

We assume an ad hoc network of N personal devices, or nodes for short, that communicate through wireless links. A node is either the personal device of a mobile user or a fixed station, as in case of a road-side gateway to/from a wired network. Thus, we are considering a hybrid urban network infrastructure [7], [15]. Throughout this paper, all nodes, either fixed or mobile, have the same capabilities; each node may act as source, recipient and forwarder of messages with specified interests. A node stores messages in a buffer, and forwards them according to the forwarding mechanism adopted.

In this paper, we consider the classical *one-to-many* communication problem to diffuse contents to a group of nodes sharing common characteristics or $behavior$. When applied to an opportunistic setting, the addressing scheme of the one-to-many communication paradigm moves apart from the IP-based addressing scheme because there is no a-priori knowledge about group identity and cardinality. This advocates the emergence of new policies in which the binding content-recipients is not provided by the sender, but directly executed by specific recipients/relays with an interest on it or in forwarding it. We name this a *behavior-driven* opportunistic addressing scheme; it underlies the most relevant one-to-many recent proposals for ONs in which the general term $behavior$ is alternatively expressed as *social-behavior* [3], *mobility-behavior* [9] or generic *interest* [11]. Hereafter we assume that each node n_x has an associated characterizing behavior $\mathcal{B}_x$. When a message labeled with $\mathcal{B}_x$ is generated by the source, the purpose of the algorithms is to deliver the message to as many nodes matching the behavior $\mathcal{B}_x$ as possible, while at the same time saving resources in terms of bandwidth and memory overhead. In this paper, we focus on forwarding policies able to deliver messages to a group of nodes sharing the attitude of visiting a (set of) common location(s), i.e., have similar *mobility-behavior*.

A. Location-based scenarios

We analyze the behavior of the two algorithms under mobility settings derived from a real dataset. To this end, the PMTR trace [6] has been adopted. It involves 44 nodes, named PMTRs (Pocket Mobility Trace Recorders), on a campus area of roughly 1000×1000 m. and equipped with wireless channel having a 10 m. radio range. There are 5 fixed nodes, located in points of interest of the campus (such as the main entrance or a cafeteria), while the remaining 39 devices have been distributed to faculties, Master and P.h.D. students, and technical staff. We eliminated nights and weekends from the complete dataset, thus producing a dataset covering 13 working days, from 8:00AM to 8:00PM (for a total of 156 hours). The PMTR dataset reproduces a difficult environment: in fact, users spend long periods in their offices, the environment is sparse, and contacts are rare [13]. In such a real scenario even the epidemic diffusion hardly performs and shows long latencies and low delivery rates [12].

We run the Louvain algorithm [1] to detect communities on the weighted graph $\mathcal{G}$ obtained from the contact trace, so that an edge exists between two nodes if one of them is a fixed location, and the other has ever met that location at least once. The edge weight is the mean duration of the contacts between the node and the fixed location. This choice allows to extract location-centered communities. We chose the Louvain algorithm as it is considered one of the best in the literature [5], it avoids grouping nodes in a giant community, and it also achieves greater modularity than other algorithms in the literature. The Louvain algorithm is supposed to detect significant communities when the modularity is greater than 0.4. From our runs we obtained a modularity of 0.536 and 5 communities (one for each PMTR in a fixed location), with minimum community cardinality of 5 nodes, and maximum cardinality of 12 nodes.

Table I
REFERENCE SCENARIOS.

scenario	**characteristics**
scenA	5 nodes in same community
scenB	5 nodes in same community
scenC	10 nodes in same community
scenD	12 nodes in same communities
scenE	12 nodes in same communities

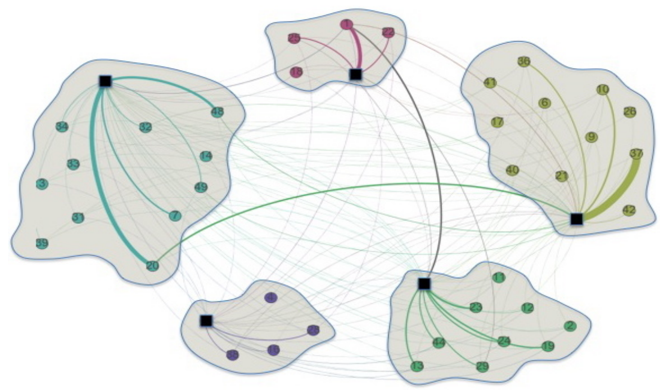


Figure 1. Location-based communities in PMTR with the Louvain method.

With the obtained communities we have defined 5 different communication scenarios, where nodes with a similar *mobility-behavior* are pooled in a *location-based* community. In Table I, we report the characteristics of the analyzed scenarios. In fig.1, the PMTR graph with communities highlighted is shown, with edge thickness proportional to the weight.[1] The fixed nodes are represented as black squares. In sec. V-A, an analysis of the characteristics of the scenarios is supplied, in terms of the characteristics of importance for the considered algorithms.

III. PROFILECAST

ProfileCast [9] has been proposed for solving the *one-to-many* problem in location-based communities.

In ProfileCast, nodes must a priori agree on the set of reference locations. For a given time slot t, each node n records the time percentage spent at each location in t, thus generating an *association vector* AV_n^t with as many entries as locations. The association matrix AM_n is then built, with each column i composed of AV_n^i, with $i \leq$ number of time slots observed so far. Hence, the association matrix represents the behavior of the node along its history. When two nodes encounter, they compare their behaviors by exchanging their association matrices. Should their behavior be *sufficiently* similar and at least one of the nodes has a message to disseminate, the message is forwarded to the other node because it is considered an interested destination. In more details, in order to save both network and computation resources, the association matrix is summarized by using *Singular Value Decomposition* (SVD) [8]. Through SVD, three matrices U, S and V are determined

[1] We emphasize that community detection is *not* performed by nodes, nor it is needed by the considered algorithms.

such that $AM = U\ D\ V^T$. The columns of U are the left singular vectors, each one representing the relative importance of the locations in the corresponding attempt of capturing the node behavior. D is a diagonal matrix of singular values, in decreasing order. The weight associated to the x-th left singular vector is computed as the ratio between the x-th singular value and the sum of all singular values. Weights aim at capturing the significance of the vectors. Let θ_W be a threshold on the weights, such that we consider that the behavior of the node is adequately described by the first v left singular vectors such that the sum of weights of those vectors is $\sum_{x=1}^{v} weight_{u_x} \geq \theta_W$. Thus, θ_W allows to determine a trade-off between the number of considered vectors and the accuracy of representation of a node behavior.

Algorithm 1 `ProfileCast`

INIT: $\mathtt{AM} \leftarrow [\ ]$; $\mathtt{AV} \leftarrow []$;
when time slot ends **do**
  for all x **do**
    $\mathtt{AV}[x] \leftarrow \mathtt{AV}[x]\ /\ \sum_{\forall y} \mathtt{AV}[y]$;
  end for
  add AV as new column to AM;
  $\mathtt{AV} \leftarrow []$;
end do
when contact with fixed location p terminates **do**
  $\mathtt{AV}[p] \leftarrow \mathtt{AV}[p]$ + duration of terminated contact;
end do
when contact with node $p \wedge$ message m held for diffusion to similar nodes **do**
  $\{U, D, V\} \leftarrow$ SVD(AM);
  for all x **do**
    $\mathtt{weight}_{u_x} \leftarrow |D[x,x]| / \sum_{\forall y} |D[y,y]|$;
  end for
  determine v s.t. $\sum_{x=1}^{v} \mathtt{weight}_{u_x} \geq \theta_W$;
  send my first v left singular vectors to p;
  receive v_p vectors from p;
  $S_{me,p} \leftarrow \sum_{M=1}^{v} \sum_{P=1}^{v_p} weight_{u_M}\ weight_{u_P^p}\ |u_M \cdot u_P^p|$;
  if $(S_{me,p} \geq \theta_{PC})$ **then**
    send m to p;
  end if
  receive messages from p and deliver them to the application layer;
end do

When two nodes n_i and n_j encounter, they exchange their first v_i and v_j left singular vectors – as determined by θ_W – respectively, and use those vectors to compute their similarity S_{ij} as follows:

$$S_{ij} = \sum_{I=1}^{v_i} \sum_{J=1}^{v_j} weight_{u_I^i}\ weight_{u_J^j}\ |u_I^i \cdot u_J^j|$$

If, let us say, n_i owns a message m to be distributed to similar nodes, and $S_{ij} \geq \theta_{PC}$ for a certain threshold θ_{PC}, then the two nodes are considered similar, and n_i forwards m to n_j. Pseudo-code of Algorithm 1 summarizes this procedure.

IV. InterestCast

ICast forwarding algorithm [11] has been designed to address the *one-to-many* communication problem for different definitions of *behavior*, and can be easily adapted to ensure content diffusion in location-based communities. In ICast, each node n maintains the list of local interests $\mathcal{I}$ and messages are tagged to characterize the content. The forwarding policy selects the good relays for reaching nodes in the set I of the users sharing the same interest, that is, the relays able to chase a given interest $\mathcal{I}$. The basic assumption is that nodes beacon their one-hop neighbors to advertise their interests and, as in [14], summary vectors are exchanged to prevent forwarding of duplicate messages. Basically, the addressing is performed on a per-content basis. The message's content-tag is matched against the declared interests of in-range nodes in order to determine its recipients. Content tags and interests do not have to match exactly. Folksonomic reasoning might be used to match nodes interests w.r.t. content tags – for instance, as described in [10] – and when a matching is verified the message is delivered to the local recipient, or forwarded to the appropriate relay. When the use of ICast is constrained to location-based interests, then the interest $\mathcal{I}$ is the identifier of one of the locations most frequently visited by the destination nodes.

Having multiple recipients that share a common interest/location $\mathcal{I}$ implies that several relays might be involved and that their selection is influenced by their attitude to encounter nodes with $\mathcal{I}$ as an interest. Miming ranking mechanisms proposed for unicast communication, we adopt a simple utility function that is adapted to reach nodes in I, rather than an addressed destination. To this end, the Greedy [4] approach has shown to obtain the best performance in several different mobility scenarios [12]. By taking inspiration from the Greedy utility, we obtain ICast, whose pseudo-code is provided by Algorithm 2. In ICast a node p adjusts its utility with respect to a given interest $\mathcal{I}$ every time it encounters a node whose beacon includes $\mathcal{I}$ (lines 3-6). In the following, let us indicate with $\mathcal{U}$ the utility value obtained with this scheme.

A relevant aspect of the algorithm is the message replication mechanism (lines 13, 18, 20). Whenever a node p, with no interest in $\mathcal{I}$, forwards a message m to a node with higher utility (lines 15-16), p delegates the other node to continue forwarding, and hence removes the copy of m from its own buffer (line 18). By contrast, if p forwards the message to a legitimate recipient, then p maintains the message copy (line 13). In fact, its habit of encountering recipients in I might be useful for delivering m to others. Nodes in I always maintain the message copy (line 20) and they can forward a copy to either another recipient (line 12) or a more useful relay which might be leveraged for delivering m to other recipients.

V. Configuring ProfileCast's parameters

As shown in the previous Sections, ProfileCast has a few parameters, namely θ_W and θ_{PC}, whose configuration is critical and may strongly influence performances. In this Section we determine the best parameter settings according to the considered scenarios.

Algorithm 2 ICast

INIT: counter $\leftarrow$ [ ]; buffer $\leftarrow \emptyset$;
when contact with node p **do**
    receive ($\mathcal{I}_p$) from p;
    send (my_$\mathcal{I}$) to p;
    counter[$\mathcal{I}_p$] $\leftarrow$ counter[$\mathcal{I}_p$] +1;
    my_$\mathcal{U} \leftarrow \{\forall$ known $\mathcal{I}$'s, counter[$\mathcal{I}$]$\}$;
    send (my_$\mathcal{U}$) to p;
    receive ($\mathcal{U}_p$) from p;

    for all messages m in my buffer **do**
        //(let $\mathcal{I}_m$ be the interest to which m is addressed)
        if ($\mathcal{I}_m == \mathcal{I}_p$) **then**
            send m to p and keep copy;
        end if
        if ($\mathcal{U}_p(\mathcal{I}_m) >$ my_$\mathcal{U}(\mathcal{I}_m)$)) **then**
            send m to p;
            if (I'm not interested in $\mathcal{I}$) **then**
                remove copy from buffer;
            else
                keep copy;
            end if
        end if
    end for
    receive messages from p and put them into buffer;
    deliver to application the messages tagged with my_$\mathcal{I}$;
end do

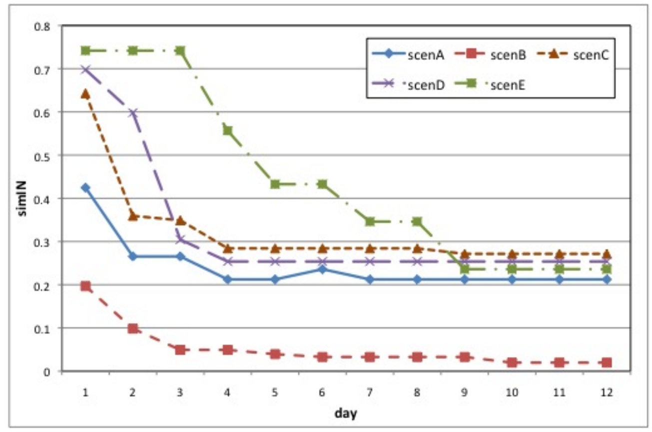
Table II
SIMILARITY BETWEEN NODES IN AND OUT OF COMMUNITY

θ_W = **0.9**	scenA	scenB	scenC	scenD	scenE
# nodes	5	5	10	12	12
simIN	0.2123	0.0197	0.2715	0.2537	0.2360
simOUT	0.2071	0.1211	0.1519	0.1703	0.1618

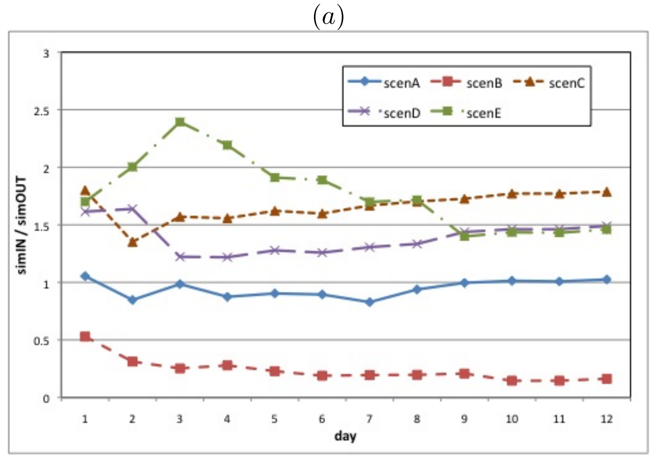
A. Analysis of the scenarios

For each scenario, we compute and analyze the similarity between either nodes belonging to the considered location-based community ($simIN$), or between nodes belonging to that community and nodes outside of it ($simOUT$). The time slot used to build association vectors is one day (i.e., 12 hours). Results are shown in Table II, where $\theta_W = 0.9$ is considered as the significant weight to determine the number of vectors adopted when computing the similarity. Mind that θ_W determines the degree of summarization of the nodes' behavior. The shown values are averaged over all possible pairs of nodes and are computed at the end of the experiment, that is, after 13 association vectors were collected. We also considered 0.7 and 0.8 as values for θ_W: in all cases, variations are negligible, in the order of 0.6% on average. As a consequence, we decided to use θ_W = 0.9 in the rest of the paper. This setting allows to exchange at most 4 vectors to compute similarity.

As it can be observed, for all, but one, scenarios the similarity amongst nodes belonging to the same community is higher than the similarity with outer nodes, thus correctly individuating communities of nodes with similar mobility behavior. These evaluations directly impact on the setting of the similarity threshold (θ_{PC}) and can have heavy influence of the algorithm performances and efficiency. The setting of the threshold θ_{PC} will be discussed in the next section.



(a)



(b)

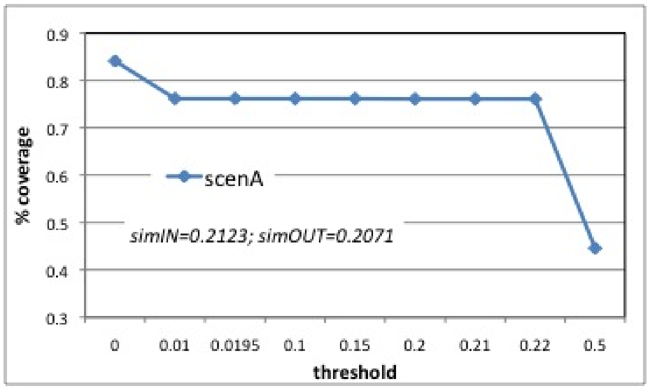
Figure 2. (a) $simIN$ along the days. (b) Ratio between $simIN$ and $simOUT$ along the days.

Interestingly, similarity shows a temporal dependency and is dynamically changing over time. As shown in fig.2(a) for $simIN$, the values of the similarity monotonically decrease along the experiment days for all scenarios, arguably denoting that users have not stable daily mobility habits. In fact in our experiment, both faculties and students may change their daily routines according to class schedules and meeting, while technical staff often displaces to offices depending on the needs. The ratio between $simIN$ and $simOUT$ stays quite constant (fig.2(b)). Yet, for the two scenarios with ratio equal or below 1 – namely scenA and scenB – it may reverse.

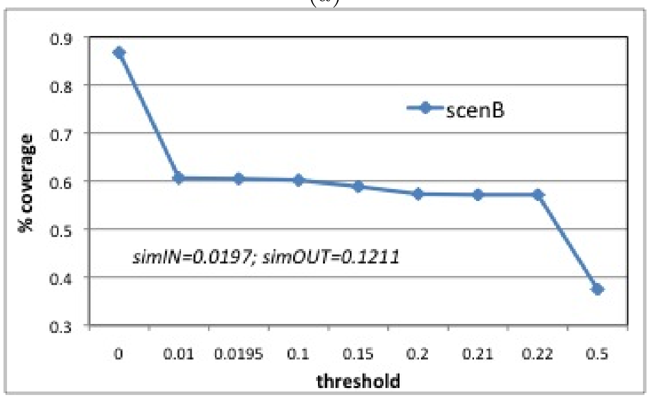
Lesson learnt: People routines may be unstable over days, thus making similarity a monotonically decreasing function. As a consequence, the setting of the value of θ_{PC} might depend on the day we are considering and on the specific location-based community to be addressed. This makes the choice of the appropriate setting very tricky because nodes have no a priory knowledge of the overall system behavior.

B. Sensitivity of ProfileCast to θ_{PC}

In this section, we analyze how the choice of θ_{PC} impacts on the performance of ProfileCast. Coherently with the setting

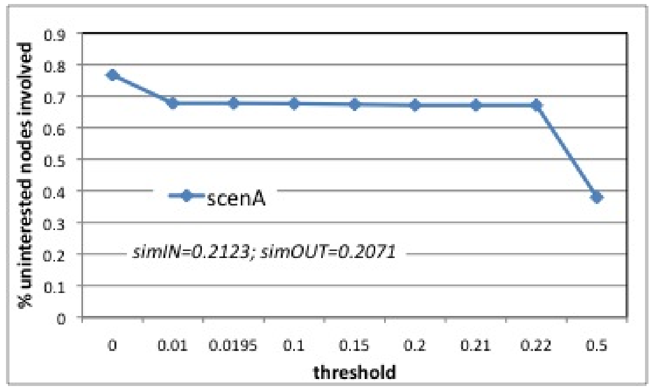


(a)

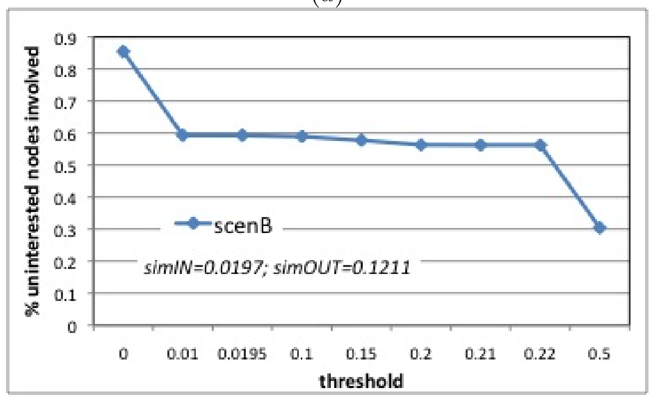


(b)

Figure 3. Mean coverage for different values of θ_{PC} in (a) scenA and (b) scenB.



(a)



(b)

Figure 4. Number of uninterested nodes receiving the message, for different values of θ_{PC} in (a) scenA and (b) scenB.

proposed in [9] we consider, for the different scenarios, $\theta_{PC} = 0.5$ and we compare the influence of this setting against a new set of values obtained from the evaluation of $simIN$ and $simOUT$, as reported in sec.V-A. In fact, the proper value for θ_{PC} should be $simIN > \theta_{PC} > simOUT$, in order to try to maximize coverage for the considered target community, while avoiding bothering uninterested nodes. The incorrect delivery of a message to an uninterested node is cause of a waste of node's resources, e.g. memory, processing, energy and radio channel. To have an indirect global estimation of such a lack of efficiency, we estimate the number of nodes receiving a message although they are not belonging to the target community and thus uninterested in the message.

In fig.3, the behavior of the coverage is shown for two different scenarios and different values of θ_{PC}. When the ratio between $simIN$ and $simOUT$ is greater than 1 (fig.3(a)) – denoting that the similarity amongst the mobility behavior of the nodes belonging to the target community is stronger than with nodes outside the community – the coverage is very stable also for values of θ_{PC} not perfectly fitting with the criteria above, that is, either slightly greater than $simIN$ or lower than $simOUT$. Nonetheless, with a higher value (namely $\theta_{PC} = 0.5$), the coverage drastically drops. Scenarios scenC, scenD and scenE behave similarly to scenA. If, by contrast, similarities are in the reverse relationship (fig.3(b)), θ_{PC} *must* be low enough to guarantee that the interested nodes are reached. The effects of this choice affect resource saving: the number of nodes not belonging to the target community and receiving the message in spite of this behaves exactly the same as coverage (fig.4). In scenB, a high coverage ($> 80\%$) is achieved when no threshold is set, thus resorting to a broadcast. In this case, 33 out of the 39 uninterested nodes – the 84$\%$ – receive the message.

Lesson learnt: as noticed in [9], θ_{PC} determines the trade-off between coverage and resource utilization. In general, the proposed value $\theta_{PC} = 0.5$ turned out to be unsuitable to perform adequately under the PMTR scenarios and a viable policy to adapt its value to unpredictable conditions seems to be hardly identified. This inability is inevitably paid with high waste of communication, energy and processing resource.

VI. Performance Evaluation

In this section, we present the performance evaluation obtained by means of simulations. Simulations recreated a variety of conditions (see, sec.II-A), where sources belong to the same location-based community C as the recipients. Message generation starts on the second day, so as to allow ProfileCast to collect at least one association vector to compute similarities. We assume that every hour, each node in the considered community generates a message labeled with an

Table III
θ_{PC} FOR DIFFERENT SCENARIOS

	scenA	scenB	scenC	scenD	scenE
θ_{PC}	0.21	0.01	0.16	0.175	0.165

attribute of a given location-based community, to be mainly used by ICast. Nodes have infinite buffers, and messages have a lifetime longer than the time needed to deliver them. Nodes encounter according to the PMTR trace, whose length is 156 hours. The performance indexes analyzed are: *coverage* (percentage of recipients in C that deliver the message), mean number of hops to reach a recipient, mean latency to a recipient, number of nodes involved as relays in the forwarding of a message. All indexes are averaged over all recipients in C and all sources. The mean number of hops and the number of involved forwarders are an indirect measure of the algorithms efficiency.

Table III shows the value of θ_{PC} we adopted for comparison between ProfileCast and InterestCast. These values are chosen in order to maximize coverage while limiting the useless message exchanges. For all the scenarios with $simIN > simOUT$ this is coincident with a value equal to $simOUT + \varepsilon$ for small ε.

With this setting, the mean number of hops followed by a message to reach its recipients – averaged over all destinations – is comparable for the two approaches, and are nearly 3-4 hops. Yet, performance is different in terms of both coverage and resource waste (fig.5). For all scenarios, ProfileCast is unable to reach the same coverage as ICast, while at the same time it delivers the messages to a number of uninterested nodes higher than the relays used by ICast. It is worth to notice that, in the case of ICast, uninterested nodes are involved in the forwarding process not because the message is erroneously delivered to them, but because they are useful bridges to reach the target community.

In fig. 6, we show the latency to reach the maximum coverage of recipients for the two protocols. The latency obtained by ICast is apparently higher than that of ProfileCast, but this is due to the fact that it is evaluated on the reached destinations, and ICast obtains a higher coverage. In fact, if we evaluate the ICast latency by truncating the message diffusion when the same coverage as ProfileCast is reached (fig. 6), we can observe that ICast takes approximately the same time to reach the same number of destinations as ProfileCast. We conjecture that for ProfileCast, the involvement of uninterested nodes – which become on their behalf message forwarders to nodes similar to *themselves* (rather than to the source) – introduces a factor of "deviation" from the path to the targeted recipients.

Lesson learnt: this experiment shows that the explicit tagging of a message, with interest name associated to the content, combined with an algorithm capable to chase interests has higher performances and efficient use of resources when diffusing contents to location-based communities. This is due to

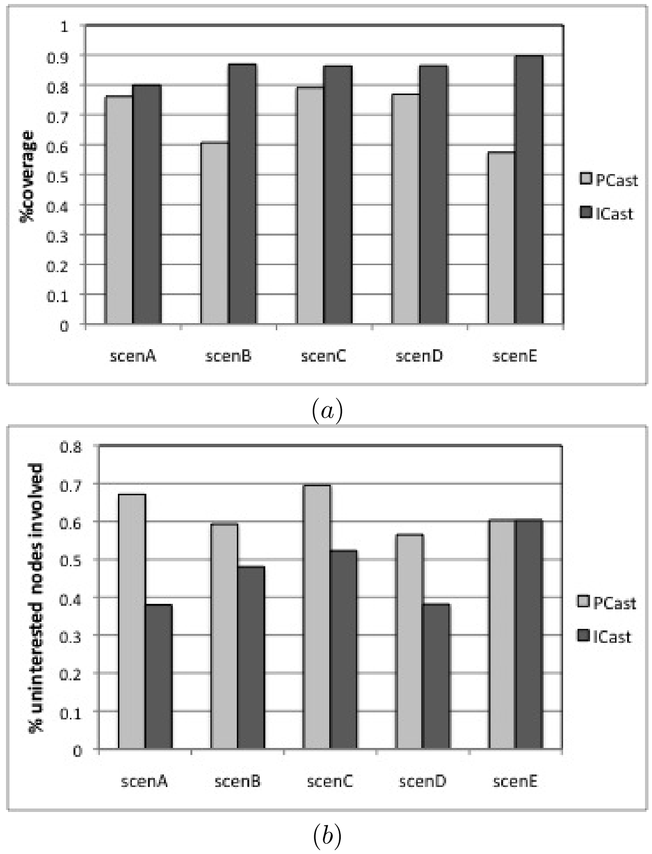


Figure 5. Comparison between ProfileCast and InterestCast in terms of (a) obtained coverage, and (b) number of uninterested nodes involved in the forwarding.

the today's inability of appropriately capturing human behavior and of adequately tuning the parameters that automatically measure the similarity between behaviors.

VII. Conclusion

In this work, we present a comparison through simulations between two mechanisms for the diffusion of messages among nodes sharing a common behavior. ProfileCast leverages mechanisms to automatically extract the intrinsic characteristics of the users from their behavior pattern; a content generated by a node is implicitly addressed to users with similar behavior as the source. InterestCast matches content tags against interests explicitly expressed by the users. In summary, ProfileCast may suffer a complex and computing-expensive parameter estimation, which depends on both time and set of target destinations. This is due to a current unavailability of accurate mechanisms for capturing and representing the manyfold aspects of user behavior. As a consequence, on one hand ProfileCast forwarding mechanism achieves lower coverage and higher latency than ICast; on the other hand, it involves a higher number of nodes not interested in the content, thus wasting resources. ICast, by contrast, does not require parametrization, it is intrinsically able to chase destinations across community boundaries, but it resorts to a rigid identification of the target, unable to identify

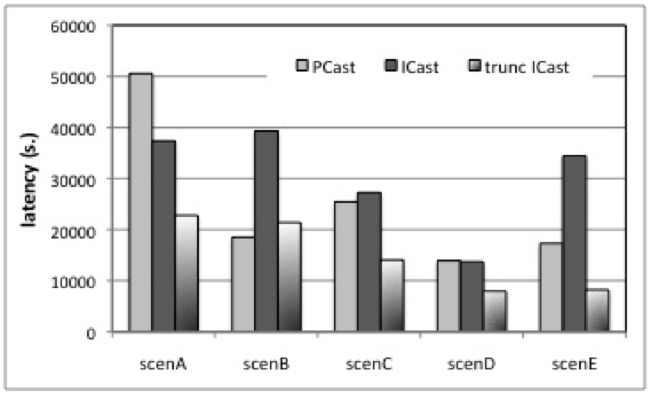


Figure 6. Comparison between ProfileCast and InterestCast in terms of latency.

destinations with affine – yet not coincident – characteristics as the source (i.e. the target).

The considered research area is very active at the moment. The main activity we are focusing on is the deployment of an in-field experiment which will enable the understanding of existing correlations among contact/sociality/location/interest of moving people. To the best of our knowledge, this would be the first dataset publicly available to the research community and will allow to validate algorithms for Opportunistic Networks in a real setting. The availability of those data will pave the way for the design of mechanisms to accurately represent users characteristics, which will be useful for inclusion in the forwarding algorithms. This analysis shall also model the dynamics of the user's behavior patterns within the considered time window. All these activities prepare the planned deployment of a testbed for one-to-many content diffusion to behavioral groups.

Acknowledgment

This work was funded partially by the Italian Ministry of Education, University and Research under the PRIN PEOPLENET (2009BZM837) Project and partially by the European Commission under the FP7 SCAMPI (258414) Project.

References

[1] V.D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, *Fast unfolding of communities in large networks*, Journal of Statistical Mechanics: Theory and Experiment 2008 (10).

[2] C. Boldrini, M. Conti, and A. Passarella, *Design and performance evaluation of ContentPlace, a social-aware data dissemination system for opportunistic networks*, J. Comput. Netw., 54(4), 2010, 589-604.

[3] P. Costa, C. Mascolo, M. Musolesi, and G.P. Picco, *Socially-aware Routing for Publish-Subscribe in Delay-tolerant Mobile Ad Hoc Networks*, IEEE J. Selected Areas In Comm. (JSAC). 26(5):748–760, 2008.

[4] V. Erramilli, A. Chaintreau, M. Crovella, and C. Diot, *Diversity of forwarding paths in pocket switched networks*, Proc. ACM/SIGCOMM IMC, 2007, 161-174.

[5] S. Fortunato, *Community detection in graphs*, Physics Reports 486, 75-174, 2010.

[6] S. Gaito, E. Pagani, and G.P. Rossi, *Strangers help friends to communicate in opportunistic networks*, Computer Networks J., 55(2), 2011, 374-385.

[7] B. Han, P. Hui, V.A. Kumar, M.V. Marathe, G. Pei, and A. Srinivasan, *Cellular traffic offloading through opportunistic communications: a case study*, in Proc. CHANTS, 2010.

[8] R.A. Horn, C.R. Johnson, *Matrix Analysis*, Cambridge University Press, 1985.

[9] W.-J. Hsu, D. Dutta, and A. Helmy, *Profile-Cast: Behavior-Aware Mobile Networking*, SIGMOBILE Mob. Comput. Commun. Rev., 12(1), 2008, 52-54.

[10] G. Lo Giusto, A.J. Mashhadi, and L. Capra, *Folksonomy-based Reasoning for Content Dissemination in Mobile Settings*, Proc. ACM CHANTS 2010, 39-46.

[11] E. Pagani and G.P. Rossi, *Interest-driven Forwarding for Delay-tolerant Mobile Ad Hoc Networks*, Proc. 9th IEEE International Wireless Communications and Mobile Computing Conference (IWCMC 2013), Cagliari (Italy), July 1-4, 2013, pp. 718-723.

[12] E. Pagani and G.P. Rossi, *Utility-based Forwarding: a Comparison in Different Mobility Scenarios*, Proc. 3rd Int'l ACM Workshop on Mobile Opportunistic Networks (MobiOpp 2012), 29-36.

[13] T. Phe-Neau, M. Dias de Amorim, and V. Conan, *Vicinity-based DTN characterization*, Proc. 3rd ACM Int'l Workshop on Mobile Opportunistic Networks (MobiOpp '12), 37-44.

[14] A. Vahdat, D. Becker, *Epidemic Routing for Partially Connected Ad Hoc Networks*. Technical Report CS-200006, Duke University, April 2000.

[15] J. Whitbeck, M. Dias de Amorim, Y. Lopez, J. Leguay, V. Conan, *Relieving the wireless infrastructure: When opportunistic networks meet guaranteed delays*, Proc. IEEE WOWMOM 2011, pp. 1-10.