On Group Mobility Patterns and their Exploitation to Logically Aggregate Terminals in Wireless Networks

Michele Rossi[†‡], Leonardo Badia[†‡], Nicola Bui[‡], Michele Zorzi[†⋆]

[†]Consorzio Ferrara Ricerche (CFR),
[‡]University of Ferrara, Department of Engineering,
via Saragat 1, 44100, Ferrara, Italy.

[⋆]DEI University of Padova,
via Gradenigo 6B, 35131, Padova, Italy.

Abstract— **Improved mobility and connectivity is one of the many challenges to be faced in future generation wireless networks. In this paper, we first propose a novel mobility model whose aim is to describe the group mobility behavior in quite a general manner. Moreover, we present a possible on–line algorithm to aggregate terminals whose links remain stable over the time, i.e., showing correlated mobility patterns. Finally, we present simulation results to validate our terminal aggregation algorithms.**

I. Introduction and Motivations

Modelling mobility in wireless networks is a challenging task. New types of mobile entities are appearing everyday, with recursive or aggregate structures moving together. One of the open issues of new telecommunication systems is to support and take advantage of these mobility patterns by improving performance whilst at the same time increasing system efficiency. That is, data retrieval by the end user should be independent of the physical location, mobility pattern and could even take advantage of the mobility itself. For example, user mobility patterns could be exploited to predict future movements and provide resource reservation in advance. More than this, users moving in a group could elect a leader, which is typically the most powerful device, and rely on this entity to route their packets. This would allow for a partial centralization of network resources with a subsequent room for increased performance. We stress that this argument is somewhat similar to what was long studied, e.g., for clustering in Ad Hoc networks, however, there are at least the following notable difference: mobility is a key issue for the presented research as we try to take advantage of correlated mobility structures to improve connectivity; the aim is to logically aggregate neighboring and stable users. A first investigation of the effectiveness of such a logical device aggregation can be found in [1], where the authors show that substantial connectivity improvements are indeed possible. However, while the authors in [1] present a study of such a device aggregation by following an analytical approach, here we focus on practical algorithms by therefore complementing their research. The intense activity to support and promote such new concepts is clearly demonstrated by the many running projects with focus on mobility. Among them, we cite here the Ambient Networks project [2].

In the current literature, several models can be found, which aim at capturing the behavior of sets of mobile entities in heterogeneous environments [3–14]. Trying to represent users' mobility is pivotal to provide good simulation tools and correctly evaluate protocol performance. A survey on mobility models proposed up to year 2000 can be found in [3], whereas the most recent solutions are discussed in [4]. Often, mobility models try to "mimic" real mobility patterns, as well as to analyze the properties of these models from a statistical point of view. The simplest mobility models are the so called *entity mobility models* [3], where users move independently of each other by following random patterns. Examples falling in this category are the Random way-point [6] and the Random Walk model [8]. More refined mobility patterns are given by *group mobility models*, like the Reference Point Group Model (RPGM) [9] and the Structured Group Mobility Model (SGMM) [12].

Mobiles terminals are now starting to possess multiple interfaces and therefore new possibilities open up for both operators and final users. In this respect, the aim of our current research is to promote terminal aggregation by possibly enhancing connectivity and, where possible, exploiting the inherent correlation of physical mobility patterns. In fact, as users usually tend to move in groups, we could think of exploiting their vicinity over the time to improve the connectivity of some of them and, in particular, of the ones experiencing bad channel conditions and/or whose radio technologies does not allow for e.g. a direct connection to a network access point (AP). In such a case, it should indeed be beneficial to elect a leader terminal, which is in the position of managing a stable communication with at least one AP and whose function will be to provide the connectivity to disconnected terminals through multi–hop communication. Clearly, these techniques involve the thorough understanding of mobility behaviors, the creation of logical structures that should be able to make the terminals aware of their close and stable neighbors and of their available resources.

The first contribution in the present paper focuses on a general framework, in which different mobility patterns can be framed. To this end, we propose a tunable model, where several "knobs" can be regulated to adjust the behavior of the terminal mobility. The strong point of the model is a compact and clear representation of the group mobility mechanics, which is described by means of attractivity between nodes without fixed constraints. Finally, other superstructures can be imposed on top of the model to make it better adhere to a specific situation, i.e., users moving along streets and/or in the presence of obstacles.

The second contribution of this work is to present a possible algorithm to perform device aggregation, i.e., to join closeby and stable users in logical groups so that they can exploit this knowledge to optimize routing decision and/or to enhance their connectivity. These groups of users will be referred to as "routing groups" (RGs) in order to differentiate them from the terminology which is commonly considered in the Ad Hoc networks case. This is done here to remind to the reader the differences between the two approaches. In a sense, one might say that the RG concept is an evolution of the clustering paradigm, aiming at dealing with multiple-technologies in a network where mobility is not only unavoidable, but can also be exploited to improve and control network performance. A simple example about the usefulness of RG structures is in the order. Consider, for instance, two nodes both in coverage of a given AP which is equipped with a medium range technology of a given type T1. Then, consider that the first terminal has both T1 and a second technology T2, whereas the second device has technology T2 only. The point of considering RGs in such a case is easily understood as if these two devices move in a group, i.e., they remain stably connected over the time, the second device could join the first one in a RG and elect it as his router to the AP. In such a case, RG structures are needed in order to gain knowledge about the surrounding network and, in turn, make decisions about the gateway to be exploited to achieve connectivity. We observe that without RGs the second device will be fully disconnected, as it only owns technology T2 while the AP is equipped with T1.

The remainder of this paper is organized as follows. In Section II, we present a novel group mobility model. In Section III we describe a distributed on–line RG formation and maintenance algorithm, whereas in Section IV we report some results on its effectiveness, by tracking the device mobility with the model in Section II. Finally, Section V concludes the paper.

II. A Novel Group Mobility Model

In the following, we outline the model basic characteristics. We consider a set of *nodes* moving around in a given area. Each node is characterized by a charge value, also referred to as *"charisma"*. A relationship of leadership is also defined among the nodes, so that each node possesses a unique *leader*, but a given node can be the leader of several nodes, called *followers*. Nodes' movements consist of two terms: a *drift movement* which tend to follow the leader and a *random movement* which can be determined by any of the available models referred above (which can be regarded as a noise superimposed to the drift component). In the next, we focus on the first term.

A high level description of the model is as follows. We consider a time-sampling of sufficiently fine granularity. At each time slot, nodes' movements are evaluated and the position updated. We consider the movement of each node to be the vector sum of two components. The first one is a randomly generated movement, whereas the second is obtained as an attraction movement toward a leader. This happens by considering a properly defined *force field*, where if C_ℓ and C_f are the charge values associated with leader and follower, respectively, we have that:

$$\overrightarrow{F_a}(d) = \begin{cases} \beta \, \dfrac{C_\ell C_f}{d^\alpha} \, \overrightarrow{u_a} & d \geq d_{min} \\ \overrightarrow{F_a}(d_{min}) & d < d_{min} \end{cases} . \quad (1)$$

In the above Equation, $\alpha, \beta \in \mathbb{R}$ are parameters used to tune the force field: α is assumed to be constant, whereas β is independently drawn, at every time step, from a Gaussian distribution with mean μ_β and variance σ_β^2. σ_β^2 is used to control the dispersion of the followers around the leader. $C_\ell, C_f \in \mathbb{R}^+$ are the nodes charges, d is the distance separating the two nodes, $\overrightarrow{F}_a(d)$ is the attraction force experienced along the axis connecting two nodes separated by d meters and $\overrightarrow{u}_a$ is the unit vector denoting this axis. Observe that we also account for a minimum distance d_{min} that is used here to limit the strength of the attraction force that, as d approaches zero would otherwise diverge. Considering nodes as particles of unit mass:

$$\overrightarrow{\Delta v_a} = \Delta t \, \beta \, \frac{C_\ell C_f}{d^\alpha} \, \overrightarrow{u_a} \quad (2)$$

where $\overrightarrow{\Delta v_a}$ is the speed variation vector associated with an interval of length Δt along the direction connecting the two nodes (follower and leader). The speed vector for a given follower at time $t + \Delta t$ is computed as:

$$\begin{aligned} \overrightarrow{v}(t + \Delta t) &= \overrightarrow{v}(t) + \overrightarrow{\Delta v}(\Delta t) \\ \overrightarrow{\Delta v}(\Delta t) &= \overrightarrow{\Delta v_a} + \overrightarrow{\Delta v_i} \end{aligned} \quad (3)$$

where $\overrightarrow{\Delta v}(\Delta t)$ is given by the sum of two contributions: the speed variation due to the attraction taking place between the follower and the leader ($\overrightarrow{\Delta v_a}$) and whose direction lies along the line connecting the two nodes ($\overrightarrow{u_a}$) and the term $\overrightarrow{\Delta v_i}$, corresponding to the speed variation associated with an independent mobility pattern, which is superimposed to the attractive behavior. $\overrightarrow{\Delta v_i}$ is obtained according to any entity mobility model. In practice, $\overrightarrow{\Delta v_a}$ is used to implement the follower-leader attractive behavior, i.e., to force the followers to move in the direction of their leader, whereas $\overrightarrow{\Delta v_i}$ is a generic entity mobility pattern, which is used to model leader–independent mobility behaviors.

As a result of our mobility modeling approach, each node tends to follow its own leader, the entity of such an attraction strongly depends on parameters α, β, C_ℓ and C_f. In addition, a random independent mobility pattern is also accounted for, by which every node tends to move according to leader–independent factors. Leaders move independently according to any entity mobility model and without being attracted by their followers. In the leader case, the speed vector at a given time $t + \Delta t$ is derived as:

$$\begin{aligned} \overrightarrow{v}(t + \Delta t) &= \overrightarrow{v}(t) + \overrightarrow{\Delta v}(\Delta t) \\ \overrightarrow{\Delta v}(\Delta t) &= \overrightarrow{\Delta v_i} \end{aligned} \quad (4)$$

It shall be observed that the group mobility model proposed here is very general as there is no need to define reference points for the followers [9]. Moreover, group movements can be adapted to very different behaviors by appropriately setting the force field ($\overrightarrow{F_a}$). Other factors, such as the presence of obstacles, streets, minimum and/or maximum distances between leader and its followers can be accounted as well through straightforward modification of the force field and by limiting the obtained mobility vectors.

For example, consider Figure 1. Here, a simple movement of a group of three nodes is depicted, in a simple scenario without

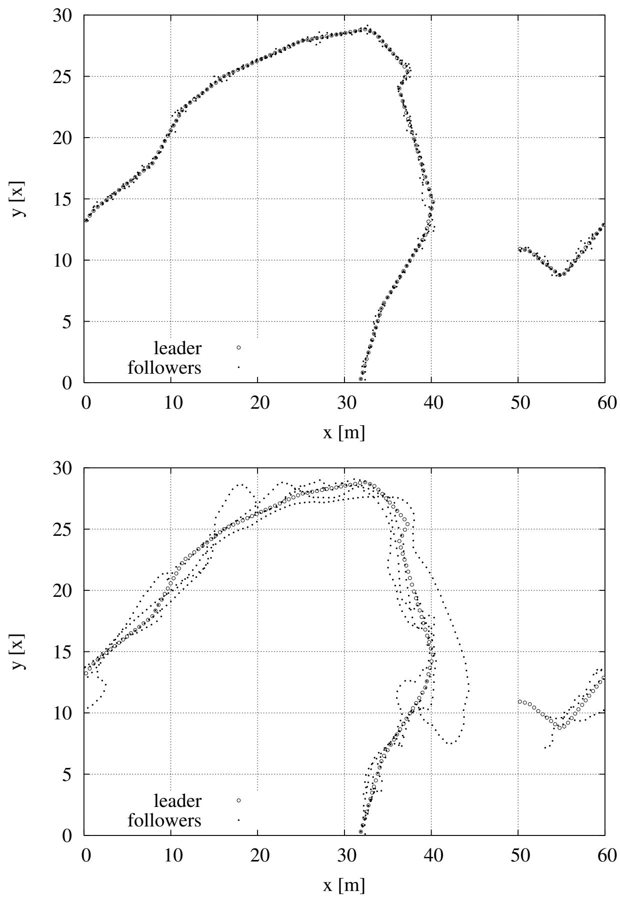

Fig. 1. Example of group mobility patterns: $\alpha = 1$, $\mu_\beta = 0.15$, $\sigma_\beta = 0.25$ (top) and $\sigma_\beta = 0.5$ (bottom), $C_f = C_\ell = 2$, $\Delta t = 1$ sec. Considered entity mobility model for leader–independent movements: boundless simulation area mobility model, where the leader moves with a constant speed of 2 km/h.

other users and no movement constraint. It is clearly visible that the group movement happens as a compact unit, though without keeping the distance among the nodes rigid or preventing users from having also movement in various (independent) directions.

III. On–Line Routing Group Creation and Maintenance Algorithms

In this section, we present an on–line algorithm for the creation and maintenance of RGs. The main point of our scheme consists of the periodic exchange of so called HELLO messages that are used to gain information about neighbors and in particular to acquire the stability of the related links over the time. This information is subsequently exploited to create logical RG structures. In the present version of the algorithm we consider that each node owns a single interface, the extension to the multiple interface case is straightforward and is objective of future research.

We first give some terminology. We refer to T_H as the lapse of time between the transmission of two subsequent HELLOs. HELLOs are sent every $T_H + \delta T$ seconds, by accounting for the random time shifts δT which are drawn from an uniform distribution. This is done with the aim of decreasing collisions as well as to avoid phase effects, i.e., the synchronization of HELLO packets transmission by multiple devices.

We introduce two further parameters, T_{SCAN} and W: T_{SCAN} is the time period considered to check and update the stability measurements (stability beliefs) of a given user, whereas W is the "window" or the maximum number of past stability measurements/HELLOs that can be memorized by a given user for any of its neighbors.

For a given user i, we account for a stability vector S_i, where we store the stability beliefs associated with any neighbor node from which user i has received at least one correct HELLO message during the last T_{SCAN} seconds. In particular, we define S_{ii} as the stability value associated with the user itself, i.e., a real number representing the stability belief of user i with respect to its first order neighbor, whereas S_{ij} is the estimated stability value for user j at user i.

Observe that while S_{ii} is a quantity representing the stability toward the set of neighboring nodes, S_{ij} addresses a specific neighbor (j). Similarly, we define a further vector R_i^0 containing the number of correct HELLO messages received by user i from any of its neighbors during the last T_{SCAN} seconds. Accordingly, $R_i^{-1}, R_i^{-2}, \ldots, R_i^{1-W}$, are the vectors related to the past $W-1$ SCAN intervals, which are separated by T_{SCAN} seconds from each other.

The algorithm works as follows: every T_H seconds (plus the random time shift) each device asynchronously sends a HELLO packet by including its current stability belief S_{ii}. A given device j, on correctly decoding this HELLO message, increases by 1 the ith entry of vector R_j^0 (R_{ji}^0).

In addition, user j modifies its stability vector S_j by updating the entry $S_{ji} \leftarrow \gamma(S_{ji} + S_{ii})$, i.e., inflating S_{ji} by the stability value just received from user i and scaling by $\gamma \in [0, 1)$. On the other hand, if user j does not correctly receive a HELLO from user i in a period of $T_H + \Delta T$ seconds, S_{ji} is updated as $S_{ji} \leftarrow \gamma S_{ji}$. ΔT is a guard interval which is obtained by summing the maximum admitted value for δT to a conservative estimate of propagation and processing delays. The reason for multiplying S_{ji} by γ is that if either the packet is transmitted and can not be correctly received or node i moves out of the radio range of user j, the stability belief S_{ji} should be reduced. In the results presented in this paper we considered $\gamma = 0.5$. The investigation of the impact of such a parameter on the performance is left for future research.

Moreover, every T_{SCAN} seconds each device i computes an auxiliary vector V_i in which it stores a measure about the quality of the link to each of its neighbors. The entry V_{ij} of V_i is calculated as a function of vectors $R_i^0, R_i^{-1}, \ldots, R_i^{1-W}$ as follows:

$$V_{ij} = \sum_{k=0}^{W-1} \left[\frac{R_{ij}^{-k}}{\eta W \lfloor T_{SCAN}/T_H \rfloor} \right], \tag{5}$$

where $\lfloor x \rfloor$ is the highest integer less than or equal to x.

First of all, note that $\lfloor T_{SCAN}/T_H \rfloor$ is the maximum number of HELLO packets that a user can send in a SCAN period. Hence, Eq. (5) will return 1 if the number of HELLOs received by user i from user j in the last WT_{SCAN} seconds equals $\eta W \lfloor T_{SCAN}/T_H \rfloor$. The parameter η is used here to tune the algorithm. After having calculated V_i, the vectors R_i^{-k} are deterministically shifted one step back and all entries in R_i^0 are re-initialized to zero.

Finally, at user i, V_i is exploited to compute the stability belief S_{ii} as follows:

$$S_{ii} = \sum_{j \in \mathcal{U}} I\{V_{ij}\} \tag{6}$$

$$I\{V_{ij}\} = \begin{cases} 1 & V_{ij} > 1 \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

where $\mathcal{U}$ is the set of users whose link was up at least once during the last WT_{SCAN} seconds. By inspecting Eq. (7) and reminding the meaning of V_{ij} one can easily verify that S_{ii} corresponds to the number of neighbors from which user i has received at least $100\eta\%$ HELLO packets out of the maximum number of receivable HELLOs during the last WT_{SCAN} seconds. Observe that in the limiting case where $\eta = 1$ S_{ii} corresponds to the number of neighboring devices from which user i has received 100% of the expected HELLO packets during the last WT_{SCAN} seconds.

The last step of the algorithm consists of the creation and maintenance of RG structures. First of all, a given user i will take part in a RG only if its stability value S_{ii} is larger than K, which roughly means that during the last scan period an equivalent number of at least K neighbors have constantly and successfully sent HELLO messages to node i. Note that varying the parameter K is a simple and effective way to modify the smallest allowable size of a RG, whereas η can be used to fine tune the stability requirements, i.e., whether a device can be considered fully stable or not. Furthermore, if $S_{ii} > K$ user i assigns a value equal to W to user j if both $V_{ij} > 1$ and $S_{ij} > 1$ and stores such a value in a further variable G_{ij}. If any of the two conditions above is not verified and $G_{ij} > 0$ then $G_{ij} \leftarrow (G_{ij} - 1)$. In this way, even if a user does not deliver a sufficient number of HELLOs such as to maintain a good stability value, it will be removed from the RG only after WT_{SCAN} seconds. This is implemented to reduce ping-pong effects due to the oscillatory behaviors of both channel and mobility. Finally, fully stable users will have a G entry equal to W, whereas other users will have a value in $\{0, 1, \ldots, W-1\}$. A given user finally decides to group with its neighbors that are stable enough, i.e., whose G entry is larger than or equal to 1, whereas a node is removed from the RG when its G entry reaches 0.

IV. RESULTS

In the following, we report some preliminary results on the effectiveness of the RG formation and maintenance scheme presented above by considering the group mobility model described in Section II.

For the mobility traces, we considered the same parameters as in Fig. 1 with the only difference that we have set $\sigma_\beta = 0.5$. Moreover, for the channel propagation we considered both path loss attenuation and Rayleigh fading. In addition, each user transmits HELLO messages in order to run the aggregation algorithms proposed above and, as two or more messages overlap at the receiver, we account for collision effects.

In the simulator, we can track ideal routing groups by inspecting the distance between nodes, their channel condition as well as their mobility patterns. An important metric that will

Average time [s]

Accuracy	$W = 1$	$W = 3$	$W = 5$
75%	13.4	20.3	29.2
90%	15.5	25.0	49.1
95%	23.6	48.7	104.0

Standard deviation [s]

Accuracy	$W = 1$	$W = 3$	$W = 5$
75%	4.2	6.0	13.6
90%	7.7	20.4	45.3
95%	36.1	64.7	96.3

TABLE I

AVERAGE LAPSE OF TIME AND STANDARD DEVIATION TO GET TO A RG ACCURACY OF $x\%$ FOR $T_H = 2.5$ S, $T_{SCAN} = 5$ S.

Average time [s]

Accuracy	$T_{SCAN} = 2.5$	$T_{SCAN} = 5$	$T_{SCAN} = 10$
75%	12.9	20.3	40.4
90%	16.3	25.0	66.8
95%	40.6	48.7	126.0

Standard deviation [s]

Accuracy	$T_{SCAN} = 2.5$	$T_{SCAN} = 5$	$T_{SCAN} = 10$
75%	3.5	6.1	14.6
90%	7.8	20.4	53.4
95%	66.4	64.7	98.5

TABLE II

AVERAGE LAPSE OF TIME AND STANDARD DEVIATION TO GET TO A RG ACCURACY OF $x\%$ FOR $W = 3$, $T_H = 2.5$ S.

be considered next consists of the degree of difference between such ideal RGs and the RGs obtained through the on–line aggregation algorithm presented above. This metric is referred here to as "RG accuracy".

Formally we define the RG accuracy as follows. Let $\mathcal{M}$ be the set of users in the simulation. Let $I_1, I_2, \ldots, I_N$ and $J_1, J_2, \ldots, J_K$ be the N groups in the ideal case and the K groups actually detected by the on–line algorithm, respectively. For every user i we define two further metrics S_i^d and S_i^p, where S_i^d is the degree of similarity between detected groups and the groups found in the ideal case, whereas S_i^p is the "potential similarity" that is defined as the maximum number of nodes belonging to both ideal and detected groups for user i.

Moreover, we consider that isolated groups give a zero similarity metric for both the detected and potential cases. On the other hand, for users belonging to non isolated groups, if $i \in I_l$ and $i \in J_m$, S_i^d corresponds to $|I_l \cap I_m|$, i.e., to the number of nodes belonging to both ideal and detected groups, whereas S_i^p is calculated as $|I_l \cup I_m|$, i.e., to the maximum number of nodes that could match between the two cases. The RG accuracy is subsequently evaluated as:

$$\text{Accuracy} = \frac{\sum_{i \in \mathcal{M}} S_i^d}{\sum_{i \in \mathcal{M}} S_i^p} \tag{8}$$

Observe that $S_i^d \leq S_i^p \;\forall\, i$ and also that when ideal and detected groups coincide for all users $\Rightarrow S_i^d = S_i^p \;\forall\, i$ and the accuracy is 1.

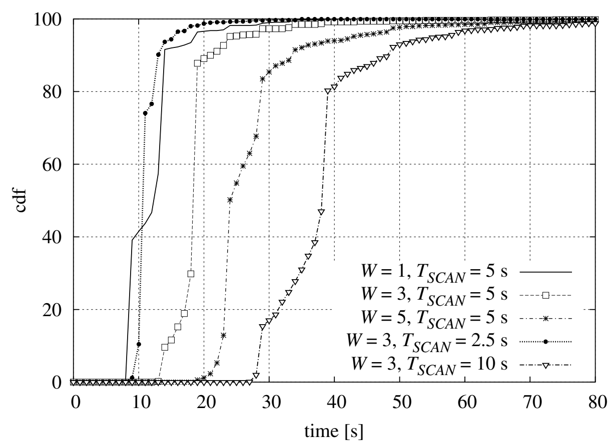

Fig. 2. Warmup phase: cumulative distribution function (cdf) of the time needed to get to a target accuracy of 75% for $T_H = 2.5$ s.

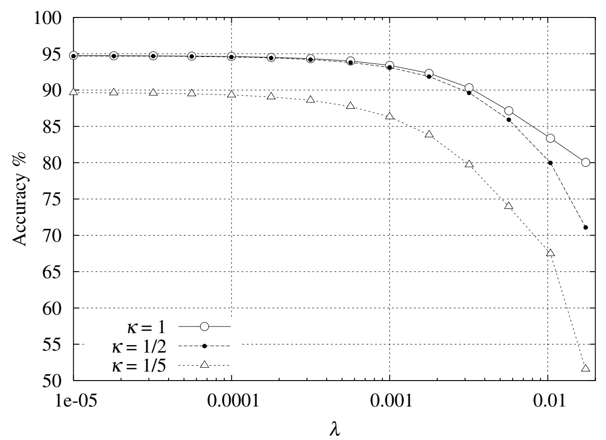

Fig. 3. RG accuracy as a function of the density of independently moving users (λ) for $W = 3$, $T_H = 2.5$ s and $T_{SCAN} = 5$ s.

In a first set of results, we focus on the RG formation aspect by considering group structures only, i.e., there are no isolated users that could somewhat affect the performance of the RG formation and maintenance scheme. In particular, in Tables I and II we consider groups of 10 users and we focus on the lapse of time needed to get to a target RG accuracy, whereas in Fig. 2 we plot the cumulative distribution function (cdf) of the time needed to get to an accuracy of 75 %.

From these results it is clear that the effectiveness of the aggregation algorithm strongly depends on the parameters W and T_H. Also, we shall observe that for small W (i.e., low complexity) and reasonable values of T_H our scheme performs satisfactorily, by detecting aggregate structures in a reasonably short time.

In addition to that, in Fig. 3 we also account for the presence of isolated users whose density is labeled as λ, these are nodes moving independently. Moreover, for the routing group sizes, we consider a factor $\kappa \leq 1$, that gives the ratio between the size of group structures and the number of isolated users within the simulation area. From Fig. 3 it is clear that the proposed solution is promising as it is also effective in filtering out spurious effects such as isolated nodes that occasionally enter the geographical regions occupied by routing groups, stationate for a while, and then leave again. Further research will be devoted to improve such an approach, which showed promising results.

V. Conclusions

This work was motivated by the trend in new generation networks for which portable devices will likely be equipped with multiple radio interfaces. In such settings, it could be beneficial to aggregate users in the so called routing groups, in order to exploit the diversity introduced by the heterogeneous scenario, by therefore improving connectivity as well as the local efficiency in the access to the channel/data forwarding. In the work presented here, we first introduced a novel group mobility model that we subsequently used to test an on-line algorithm, also proposed in the paper, for the creation and maintenance of routing groups structures. The results are encouraging in the sense that even a very light exchange of information between nodes allows for a proper recognition of aggregated structures in a reasonable amount of time by using on-line and fully asynchronous schemes. Our future research is targeted in the extension of the presented scheme for the multiple-interface case and in testing it for different mobility patterns and radio characteristics.

Acknowledgment

This paper describes work undertaken in the context of the Ambient Networks project, which is part of the EU's IST program. In total 41 organizations from Europe, Canada, Australia and Japan are involved in this Integrated Project, which will run from 2004-2005 in its first phase. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the Ambient Networks Project.

References

[1] M. Rossi, L. Badia, P. Giacon, and M. Zorzi, "On the effectiveness of logical device aggregation in multi-radio multi-hop networks," in *Proceedings of IEEE MobiWac 2005*, Maui, Hawaii, US, June 2005.

[2] "The Ambient Networks Project." [Online]. Available: http://www.ambient-networks.org

[3] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wiley Wireless Communications & Mobile Computing (WCMC)*, vol. 2, no. 5, pp. 483–502, Aug. 2002.

[4] Q. Zheng, X. Hong, and S. Ray, "Recent advances in mobility modeling for mobile ad hoc network research," in *Proceedings of the 42nd ACM annual Southeast regional conference*, Huntsville, Alabama, US, Apr. 2004.

[5] C. Bettstetter, "Smooth is better than sharp: a random mobility model for simulation of wireless networks," in *Proceedings of ACM MSWiM 2001*, Rome, Italy, July 2001.

[6] D. B. Johnson and D. A. Maltz, *Mobile Computing*, T. Imielinski and H. Korth, Eds. Kluwer Academic Publishers, 1996, Chapter 5: Dynamic source routing in ad hoc wireless networks.

[7] E. Royer, P. Melliar-Smith, and L. Moser, "An analysis of the optimum node density for ad hoc mobile networks," in *Proceedings of IEEE ICC 2001*, Helsinki, Finland, June 2001.

[8] M. Zonoozi and P. Dassanayake, "User mobility modeling and characterization of mobility patterns," *IEEE JSAC*, vol. 15, no. 7, pp. 1239–1252, Sept. 1997.

[9] X. Hong, M. Gerla, G. Pei, and C.-C. Chiang, "A group mobility model for ad hoc wireless networks," in *Proceedings of ACM MSWiM 1999*, Seattle, Washington, US, Aug. 1999.

[10] A. Jardosh, E. Belding-Royer, K. Almeroth, and S. Suri, "Towards realistic mobility models for mobile ad hoc networks," in *Proceedings of ACM MobiCom 2003*, San Diego, California, US, Sept. 2003.

[11] X. Hong, T. Kwon, M. Gerla, D. Gu, and G. Pei, "A mobility framework for ad hoc wireless networks," in *Proceedings of International Conference on Mobile Data Management (MDM 2001)*, Hong Kong, Jan. 2001.

[12] K. Blakely and B. Lowekamp, "A structured group mobility model for the simulation of mobile ad hoc networks," in *ACM MobiWac 2004*, Philadelphia, Pennsylvania, US, Oct. 2004.

[13] Z. Haas, "New routing protocol for the recongurable wireless networks," in *Proceedings of ICUPC 1997*, San Diego, California, US, Oct. 1997.

[14] B. Liang and Z. Haas, "Predictive distance-based mobility management for PCS networks," in *Proceeding of IEEE INFOCOM 1999*, New York, NY, US, Mar. 1999.