Weighted Centrality by Potential for Replica Relocation in MANETs

Gisane A. Michelon[1], Luiz A. P. Lima Jr.[2], José Aélio de Oliveira Jr.[2], Gil E. de Andrade[2]

[1] UNICENTRO/Department of Computer Science, Guarapuava, Brazil.
[2] PUCPR/Post-Graduate Program on Computer Science (PPGIa), Curitiba, Brazil.

* Corresponding author. Email: gil.andrade@ifpr.edu.br

Abstract: Mobile Ad Hoc Networks (MANETs) are characterized by several dynamic factors (e.g., constant topology changes and limited node resources) which deeply affect data availability on the network. Therefore, a major challenge in this context is to develop autonomous adaptive application-dependent methods for data replication. This paper presents a distributed centrality algorithm based on an application-level message forward mechanism called Virtual Magnetic Fields that adjusts the center of the network graph according to dynamic factors such as data access frequency and node stability, while tackling changes in the network topology. By presenting and analyzing simulation results, this paper shows that the relocation of replicas of data based on the proposed algorithm significantly reduces the overall network traffic required to access those data.

Key words: Centrality, data replication, MANETs, virtual magnetic fields.

1. Introduction

Mobile Ad-hoc Networks (MANETs) are infrastructure-less wireless networks in which nodes do not have a prior knowledge of the global network topology. The nodes of a MANET act as routers and communicate with each other by forwarding packets through multiple hops [1]. Due to the highly dynamic nature of MANETs, data accessibility is a real challenge in this context. Information may be available on one node at one moment and, just after that, this node may move away from another requesting node or may just become inaccessible.

Therefore, data need to be relocated as the network topology changes and context information (such as *access frequency* and *node stability*) becomes then crucial for deciding which node is the most appropriate to hold a replica of those data. Moreover, the choice of replica location has a big impact on the data availability. In MANETs, the problem of the replica location is more difficulty to be resolved. Often, it not feasible to use algorithms that produce exact solutions, being proposed approximation algorithms, that find solution close to the optimal for a given time [2].

In order to conceive a general and dynamic model of replica placement, an appropriate paradigm is needed. In this work is used of concept of *Virtual Magnetic Fields* (VMFs) [3]. This paradigm originally allows message forwarding through the construction of overlay networks representing *attraction relations* among nodes, so that messages are "attracted" towards the target nodes regardless of their location in the network [3]. The messages that arrive at a node are forwarded towards the node that exerts the greatest magnetic influence (or "attraction force") according to prearranged magnetic relations [4]. The hypothesis

followed was that VMFs can be adapted to work in a mobile environment and that replicas can be attracted to the best node determined by dynamic factors. By doing so, a solution to one of the main problems of data replication is found, namely, maximizing data availability in the network while minimizing the use of device and network resources. It was not found other works in literature that used this approach (e.g., considering dynamic factors).

This paper then addresses the problem of placing replicas at the "best" nodes by using a *Weighted Centrality by Potential* that takes into account the inherent dynamicity of MANETs. A new method is therefore proposed for data replication based on the concept of VMFs and considering data access frequency and node stability. This information is used to compute the node's potential, i.e., how intensely this node is attracting replicas to itself. Since our only concern is with data relocation, data are assumed to be used for read-only purposes. Therefore, replica synchronization is out of the scope of this work.

The remainder of the paper is organized as follows. Section 2 discusses related work before presenting the proposed replication method and *Weighted Centrality by Potential* in Section 3. Section 4 shows the simulation results. And the Section 6 draws conclusions and points at future work.

2. Related Work

Hara [5] was among the pioneer in the area of data replication in MANETs. Hara's model uses three main techniques, namely, SAF (Static Access Frequency), DAFN (Dynamic Access Frequency and Neighborhood) and DCG (Dynamic Connectivity based on Grouping). DCG does not allow group division even when a node gets disconnected. The latter is the most complex among the three [4]. DCG groups the mobile nodes in such a way that the group is not divided even if a node disconnects from the network. The coordinator group is responsible for determining the allocation of replicas based on the access frequency obtained from its members, and then, it allocates the replica in the node that has the highest number of accesses to the corresponding data item. However, Hara's work does not consider changes in the network topology in order to relocate replicas and is, therefore, unsuitable for networks with mobility [4]. Because the relocation of data is proactive and the relocation time interval is fixed, another problem arises, namely, how to define this interval. If it is too large, data access may be compromised. If it is too small, the data traffic can be very high.

SCALAR [6] is another distributed data replication framework. In SCALAR, the network nodes form a virtual backbone using a connected dominating set (CDS) based on the network topology graph. The goal is to improve the accessibility of data and to reduce the overhead of network communication. The simulation results were then evaluated so that the extra overhead added by protocol, the data accessibility, and other important factors for MANETs could be assessed. The performance of SCALAR was then compared with other two well known techniques [6]: the SAF and DAFN already mentioned above. SCALAR performs better than the other two techniques in relation to data accessibility, messaging overhead and scalability of the network, especially for networks with many nodes. But the decision of replicating data items is based solely on the request frequency of an item and the distance of the received data's owner [6]. SCALAR does not consider the instability of the nodes. So, even when the speed of the node is high, or its battery level is low, or yet it has limited memory or processing capacity, the node may still store a replica.

In EARAM-SN [7], a distributed algorithm is proposed for adaptive allocation of replicas in order to place a replica as near as possible to the optimal solution. This method for replica allocation considers the access frequency of read/write operations and network topology in order to minimize access costs. The access frequency statistics are collected only for stable neighbors. The stability of the neighborhood is defined by the distance between two neighbors (that can be measured by GPS) in a time frame. Thus, the oscillation of replica allocation is reduced when the mobility of nodes is high.

In EVC [8], the authors used the centrality eigenvector of Bonacich [9]. Unlike of the degree centrality, EVC does not assume that each connection to a node is equally important. The idea is that the most

important node should have the greatest influence on the calculation of centrality. The centrality of a node depends on the number of connections and on the connection quality (weight of a link) in relation to the frequency of connections to other nodes. The authors show that the results obtained, in terms of data accessibility, by allocating the replica using EVC were better than those when a random node was chosen. However, when the number of nodes in the network increased, EVC did not improve data accessibility.

In [10], a method was developed for predicting replica mobility so that data can be accessed by any node in a minimum number of hops. The accessibility of data increases because nodes that may cover a larger number of other nodes in a smaller number of hops are selected on the network. Simulation showed that the response time and the delay to access the data has been reduced. However, the two algorithms used by the authors are computationally complex. The time complexity for computing the minimum dominant set is $O(2^n)$ and for computing the subgraph centrality is $O(n^3)$. Exponential time algorithms are only useful for small number of nodes [11]. Table 1 presents a summary of the related work.

Table 1. Comparison of Replication Methods

Method	Considers	Does not consider
Hara	Access	Instability and number of hops
SCALAR	Access Distance to the data	Instability
EARAM	Access Instability (in relation to mobility)	Number of hops
EVC	Centrality (in relation to connectivity)	Access and number of hops
Subgraph Centrality	Centrality (in relation to hops number) Instability (in relation to connectivity)	Access Algorithm complexity

3. System Model

The model of data replication is reactive in the sense that changes in the environment are taken into account and that the decisions regarding replica relocation autonomously adapt to the context. The decision is made considering: (a) the *access frequency* to the data; (b) the *distance* in hops to access the data; and (c) the *node instability*. The latter is typically computed from node's degree of mobility, its connectivity (the rate at which the node connects and disconnects in a short period of time), its transmission range (that characterizes the scope coverage) and the amount of resources available (remaining battery level, node storage and processing power) [4].

This reactive model allocates data using contextual parameters dynamically obtained in order to generate less network overhead based on how frequently they are requested. The goal is to relocate the replica closer to the nodes that access it more frequently, but taking into account the degree of stability of the surrounding nodes so that replicas tend not to be placed on unstable nodes [4]. This approach obviously aims at reducing the communication costs and delay [4]. In order to deal with changes in the environment, such as node speed and the availability of the node's resources, the concept of virtual magnetic fields [3] will be used, as mentioned in Section 1. Only the nodes that are direct neighbors of the node that has the replica (one hop) have an attraction force value (or "*potential*") computed from the current context information stored in each node. Moreover, the attraction force of a node is computed within a time frame, avoiding the data replicas from moving towards nodes that accessed them a long time ago [4]. The access frequency measure decreases along a time interval *Δt* when the replica is no longer accessed. The node instability is also considered within *Δt*, after which instability needs to be updated.

Let $G = (V, E_t)$ be the connected graph representing the network topology at time *t*. *V* is the set of nodes and E_t is the set of undirected edges. Let $n = |V|$ be the number of nodes in the graph and $m = |E_t|$, the

number of edges at instant *t*. We denote $s(i, j)$ as the shortest distance in hops between nodes *i* and *j* ($i, j \in V$) and $N_t(i)$ is the set of direct neighbors of node *i* at moment *t*. Let *R*, be the set of nodes that hold data replicas at a given moment ($R \subseteq V$). The method consists in weighting the centrality taking into account dynamic factors such as the access frequency and instability of the nodes in a MANET. This weighted centrality is quantified by a *potential value* which combines a *centrality measure* (based on the *localized volume-based centrality – Vol* [12] – an approximation of the closeness centrality) and a measure of *dynamic factors* that may affect the estimation of the best node to store the data at some point in time. The *potential* of some node *i* corresponds then to the strength with which node *i* "attracts" replicas to itself. The potential $P(i)$ of a node $i \in V$ is given by (1).

$$P(i) = \alpha \cdot Vol(i)_{Norm} + (1-\alpha) \cdot W(i) \tag{1}$$

The first portion of (1) is related to the centrality as computed from $Vol(H^i_h)$, but normalized so it will stay within the interval [0,1]. $Vol(H^i_h)$ corresponds to the sum of the degrees of all the neighbors of the node *i*, whose distance from *i* is less than or equal to *h*-hops. $Vol(i)_{Norm}$ is given by (2).

$$Vol(i)_{Norm} = \frac{Vol(H^i_h)}{n \cdot (n-1)} \tag{2}$$

Since *n*-1 is the maximum number of links that node can have, then $Vol(H^i_h)$ is limited to *n*(*n*-1) (for *h*=1).

The second portion of (1) expresses the dynamic factors gathered under weight $W(i)$ ($W(i) \in [0,1]$)*.* The dynamic factors considered are the *Access Frequency – AF* and the node *Instability – IN*. $W(i)$ is the *weight* to be applied to node *i*, which is proportional to the access frequency *AF(i)* and inversely proportional to the instability measurement *IN(i)* (*IN(i) 1*) as defined by (3). Notice that, the more frequently replica access requests are issued (or forwarded) by node *i*, and the more stable it its, the greater $W(i)$ will be.

$$W(i) = \frac{AF(i)}{IN(i)} \tag{3}$$

Coefficient *α* (0≤*α*≤1) is a *weighting factor* introduced with the purpose of adjusting how intensely the centrality and the weight *W* will affect the computed potential. The precise value of *α* is defined by the application and must be a global knowledge of all the nodes. The greater the value of *α*, the less the dynamic factors will influence $P(i)$. Notice that $P(i) \in [0,1]$.

Two different equations have been defined to compute the access frequency ($AF(i)$) of some node *i*. If $i \in V - R$ (i.e., *i* is not currently storing a replica), the access frequency is computed from the number of replica access requests the node issues itself or passes on behalf of other nodes. On the other hand, if $i \in R$, then its access frequency needs to be attenuated, otherwise, nodes in *V-R* would have much smaller access frequency rates and, therefore, would not possibly attract the closest replica to themselves. So, in order to compute the access frequency of node *i*, it is necessary to distinguish between *local* access requests (i.e., those issued by node *i* itself) and *remote* access requests (i.e., those issued by *i* on behalf of other nodes). And if only the local access of node *i* was considered, then its potential would be smaller and the replica could move unnecessarily. Equation (4) is used to compute $AF(i)$, when $i \in V - R$.

$$AF(i) = \frac{AC(i)}{\max AC} \tag{4}$$

$AC(i)$ is the Access Count, i.e., the number of local and remote requests issued by node i in the last Δt time units. max AC is the maximum number of accesses in Δt supported by the system (max AC is considered a globally known value). Notice that $0 \leq AF(i) \leq 1$.

$AF(i)$, therefore, designates the number of times node i has accessed the replica in the last Δt time units. Local and remote access requests increment $AC(i)$.

If $i \in R$, $AC(i)$ is not incremented when the replica stored in i is accessed by another node $j \neq i$. This restriction is necessary because node i is not technically on the "path" of a replica. Since node i concentrates access requests from other nodes, if $AC(i)$ was incremented in each access, then $AC(i)$ would be at least as large as $AC(j)$ (for $j \in N_t(i)$). And since $P(i)$ is dependent on $AF(i)$ which is directly proportional to $AC(i)$, then node i would hardly loose its replica to a neighbor. Let $AC_N(i)$ be the maximum access count of a direct neighbor of node i, as expressed by (5).

$$AC_N(i) = \max\left\{AC(j) : j \in N_t(i)\right\} \tag{5}$$

Equation (6) is used to compute $AF(i)$, when $i \in R$.

$$AF(i) = \frac{\beta(i) \cdot AC_N(i)}{\max AC} \tag{6}$$

$AF(i)$ is here a function of $AC_N(i)$ weighted by difference between $AC_N(i)$ and the local access request count ($AC(i)$) represented by coefficient $\beta(i)$. This difference is designated by (i), as show in (7).

$$\delta(i) = AC_N(i) - AC(i) \tag{7}$$

$\beta(i)$ – (8) – is necessary to compensate the fact that the node that has the replica concentrates naturally a greater number of access as explained above.

$$\beta(i) = 1 - \frac{k \cdot \delta(i)}{(k-1) \cdot |\delta(i)| + \max AC} \tag{8}$$

The node that stores the replica may not have any local access of its own. If only $AC_N(i)$ is used, node i would likely have a larger $AF(i)$ and consequently a potential $P(i)$ larger than all of its neighbors. If only local $AC(i)$ is considered, the replica could be moved unnecessarily to some other node. The graph of $\beta(i)$ is shown in Fig. 1 for max AC=100. If the maximum access count of a direct neighbor is larger than local access count,
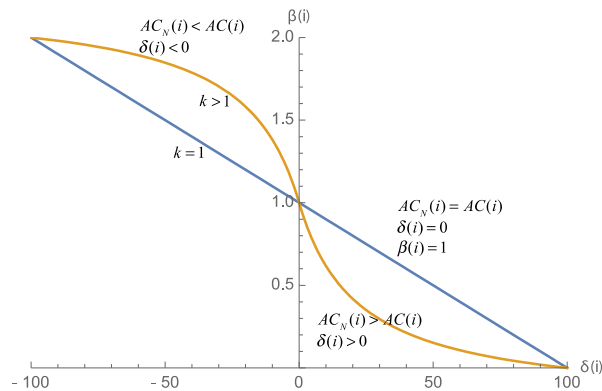

Fig. 1. Coefficient $\beta(i)$.

k is a constant used to adjust how intensely $\delta(i)$ will affect $\beta(i)$. When $k = 1$, the decay is linear. k influences how fast a replica will move to some other node. Notice that $\delta(i) \leq \max AC$ because $\max AC \geq AC_N(i)$ and $\max AC \geq AC(i)$. Moreover, if AC_N is equal to $AC(i)$, then $\delta(i)$ is zero, $\beta(i)$ will be 1 and $AF(i)$ will be calculated as when $i \in V - R$.

When a node $i \in R$ looses its replica to a neighbor (being removed from R), its local access count ($AC(i)$) needs to be updated so that the i's potential does not raise drastically, causing the replica to immediately come back to it. While it had the replica, the calculation was dependent on the access count of its direct neighbors. But, when it loses the replica, a new value needs to be assigned do $AC(i)$ so that the node neither attracts back the replica nor drops its potential to much delaying replica attraction towards nodes on the opposite side of the neighbor that now has the replica. Let $P(i)$ be the potential of node i when it belongs to R and $P'(i)$ the new potential after losing the replica. In order to prevent the replica from coming back to i, $P'(i)$ must be, at most, $P(i)$. From this, the new value of the local access count – $AC'(i)$ – when i looses the replica, may be deduced (see (9)), assuming that $IN(i)$ has not changed and $k=1$.

$$AC'(i) = \left\lfloor \left(1 - \frac{\delta(i)}{\max AC}\right) \cdot AC_N(i) \right\rfloor \tag{9}$$

After computing $P(i)$, an algorithm runs on nodes in R to determine if a replica needs to be relocated or duplicated. A node r in R will pass the replica on to another neighboring node i only if $P(i) - P(r)$ is greater than a threshold τ. If there are two nodes i and r so that $P(i) - P(r) > \tau$, a copy of the replica is created and moved to the node with a larger potential (this node will attract the replica in its direction). When two or more replicas reach the same node, just one of them is kept. If no requests are issued for relatively long period of time, the dynamic portion of the potential in (1) will drop to zero and the replicas will tend to go to the center of the network being automatically destroyed. The value of τ is estimated from the cost of sending a replica along one hop in the network. τ may change depending on specific characteristics of the network and application.

4. Simulation and Performance Analysis

The proposed method for data replication was simulated through the OMNeT++ framework [13]. The main simulation parameters used are shown in Table 2.

Table 2. Simulation Parameters

Parameter	Value
Area	300m $\times$ 300m [6], 450m $\times$ 450m and 600m $\times$ 100m
Node speed	0-3 m/s [6] and 0-6m/s
No. of mobile nodes	20[6]
Mobility model	Random WayPoint (RWP) [6]
Communication range	100m [6]
Size of a data item e size message	8000B e 4B

Fig. 2 and Fig. 3 show, respectively, the average number of hops required and the message load in KB sent per request using Weighted *Centrality by Potential* (*WCP*) in comparison with a random choice ("*No centrality*") and static centrality ("*Vol*") of the nodes that will store the replica. In this scenario, only two nodes (among the 20) request access to the replica. In order to better observe replicas getting close to requesting node, the number of replicas allowed in this scenario was fixed to two.

From Fig. 2 and Fig. 3, one can notice that, using WCP, the replica traverses a smaller number of hops to

get to the requesting node when compared with the "*No Centrality*" approach. Moreover, WCP causes a decrease in the number of hops that are needed to access the replica as the number of requests increases. However, the number of hops remains nearly constant when using a random choice. As expected, the number of hops decreases with the WCP method with the increasing number of requests because the replicas are "attracted" to the node that has a higher potential strength. And this potential is influenced by the *AF*.

Fig. 3 shows the data volume in KB sent per request, including the control data exchanged for computing *Vol* and the *Weighted Centrality by Potential*.
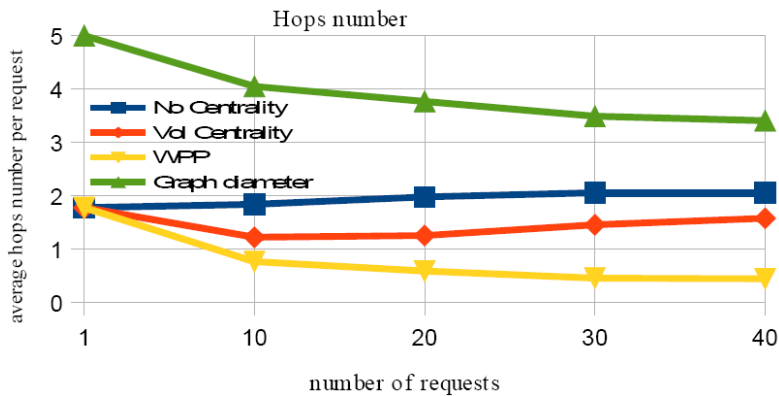

Fig. 2. Average number of hops required.

The performance of *WCP* was then studied when the speed of the nodes increased from the range 0-3m/s to 0-6m/s. The goal was to investigate the impact of the network topology changes in the overall performance. Fig. 6 shows the results of the simulation in this scenario, comparing speeds of the range 0-3m/s with those of the 0-6m/s. Smaller speeds produce, obviously, better results. A strategy to improve the performance of *WCP* when nodes move faster is to decrease α, so that the expensive computation of the centrality (needed by *P(i)*) could be performed less frequently. Notice, however, that the network centrality does not change drastically. So, an outdated centrality still remains a good approximation for a longer time.
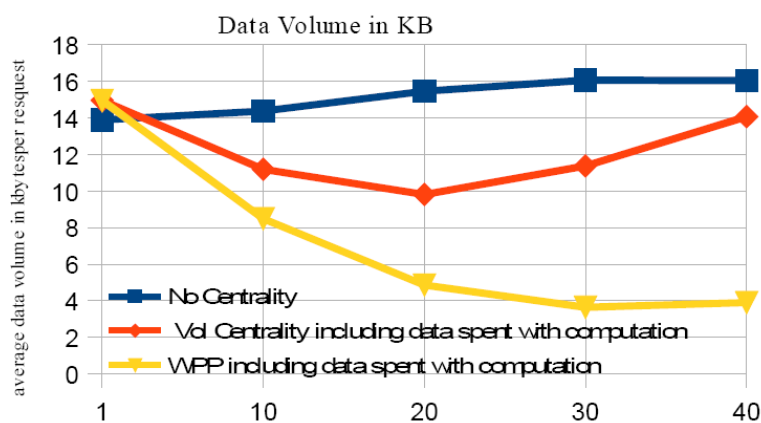

Fig. 3. Byte load per request (2 nodes).

The performance of WCP was then studied when the speed of the nodes increased from the range 0-3m/s to 0-6m/s. The goal was to investigate the impact of the network topology changes in the overall performance. Fig. 4 shows the results of the simulation in this scenario, comparing speeds of the range 0-3m/s with those of the 0-6m/s. Smaller speeds produce, obviously, better results. A strategy to improve

the performance of WCP when nodes move faster is to decrease α, so that the expensive computation of the centrality (needed by $P(i)$) could be performed less frequently. Notice, however, that the network centrality does not change drastically. So, an outdated centrality still remains a good approximation for a longer time.

Fig. 5 shows the results of a simulation run with α =0,20 and a mobility of 0-6m/s. The influence of *Vol* in the computation $P(i)$ is therefore decreased and W (representing the dynamic factors) will impact more intensely the computation of the node potential. However, problem may happen if the centrality is disregarded: at some point, a replica may be located at some node which is very far from another requesting node (one of the goals of considering the centrality in the computation of $P(i)$ is to keep the replica at a minimum distance from all nodes). The other side effect is that replicas will not probably be automatically destroyed when no requests are issued for a long time.
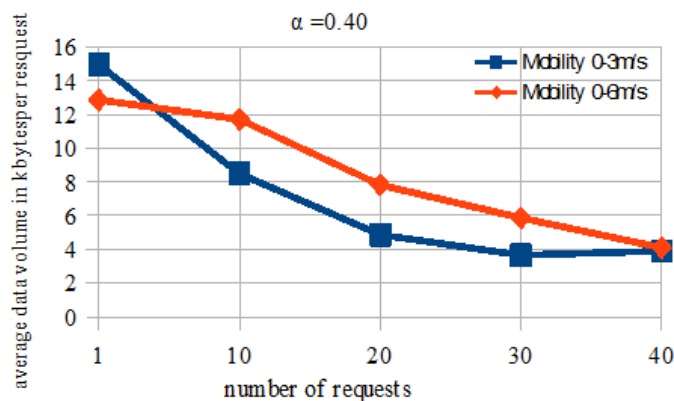

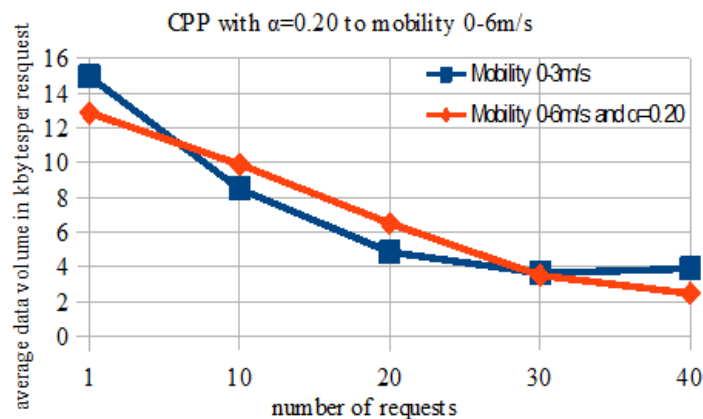Fig. 4. Increased node mobility with α=0.40.

Fig. 5. Increased node mobility with α=0.20 and mobility speed of 0-6m/s.

Finally, notice that even if the centrality is computed regularly as the network topology changes, the Vol algorithm only needs to run on the nodes in the path from the requesting node to the node that stores the replica.

5. Conclusion

Although some related work can be found in the literature, it were not found a method for dynamic replica placement in MANETs using the same strategy as WCP. After computing the potential of each node, this information is used to make decisions about the creation, relocation and destruction of replicas. In order to reduce the number of hops required to access a replica, the proposed method tries to bring the replica as near as possible to the nodes that access it more frequently, avoiding, at same time, unstable

nodes. This is done, however, without completely overlooking the graph center, since any other node may, sooner or later, request the data too. The equations presented to compute the potential of the nodes were designed to displace the network "physical" center by taking into account dynamic factors (access frequency and nod stability, in our case).

The simulation results have shown that the method defined reduces in 70%, in the average, the number of bytes exchanged by the nodes when the number of requests raises above a certain threshold in the scenarios investigated. A deeper study on the impact of coefficients such as α, Δt, τ and k is however still considered future work. Moreover, other dynamic factors present in MANETs may also be included. The impact of mobility in the performance of WCP in specific scenarios such as vehicular networks still need to be assessed. In any case, it is believed that the WCP method based on the Virtual Magnetic Fields paradigm is viable and original.

Acknowledgment

References

[1] Hadim, S., Al-Jaroodi, J., & Mohamed, N. (2006). Middleware issues and approaches for mobile ad hoc networks. *Proceedings of 3rd IEEE Consumer Communications Networking Conference*: *Vol. 1* (pp. 431-436).

[2] Jing, Z., Jinshu, S, Kan, Y., & Yijie, W. (2004). Stable neighbor based adaptive replica allocation in mobile ad hoc networks. *Computational Science (ICCS)*, *3036*, 373-380.

[3] Lima, L. A., & Calsavara, A. (2010). Autonomic application-level message delivery using virtual magnetic fields. *Journal of Network and Systems Management, 18(1)*, 97-116.

[4] Hara, T., & Madria, S. K. (2009). Consistency management strategies for data replication in mobile ad hoc networks. *IEEE Transactions on Mobile Computing, 8(7)*, 950-967.

[5] Michelon, G. A., Lima, L. A. P., De Oliveira, A. J., Calsavara, A., & De Andrade, G. E. (2014). A strategy for data replication in mobile ad hoc networks. *Proceedings of IEEE 22nd International Symposium on Modeling, Analysis & Simulation of Computer and Telecommunication Systems* (pp. 486-489).

[6] Atsan, E., & Ozkasap, O. (2013). SCALAR: Scalable data lookup and replication protocol for mobile ad hoc networks. *Computer Networks, 57(17)*, 3654–3672.

[7] Jing, Z., Jinshu, S., Kan, Y., & Yijie, W. (2004). Stable neighbor based adaptive replica allocation in mobile ad hoc networks. *Computational Science (ICCS), 3036*, 373-380x.

[8] Atsan, E., & Ozkasap, O. (2007). Applicability of eigenvector centrality principle to data replication in manets. *Proceedings of 22nd International Symposium on Computer and Information Sciences* (pp. 1-6).

[9] Bonacich, P. (1987). Power and centrality: A family of measures. *American Journal of Sociology*, *92(5)*, 1170-1182.

[10] Pushpalatha, M., Ramarao, & Venkataraman, R. (2014). Applicability of subgraph centrality to improve data accessibility among peers in MANETs. *Peer-to-Peer Networking and Applications, 7(2)*, 129-146.

[11] Cunha, I. (2012, December) *Complexity Analysis. Federal University of Minas Gerais*. Retrieved October, 11, 2014, from http://homepages.dcc.ufmg.br/~cunha/teaching/20121/aeds2/complexity.pdf

[12] Wehmuth, K., & Ziviani, A. (2012). Distributed assessment of the closeness centrality ranking in complex networks. *Proceedings of the Annual Workshop on Simplifying Complex Networks* (pp. 43-48).

[13] Varga, A., & Hornig, R. (2008). An overview of the OMNeT++ simulation environment. *Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems & Workshops* (pp. 1-10).

Gisane A. Michelon was born in Fraiburgo, Brazil, in 1981. She received the degree in analist of systems in 2004 from University of Unicentro from the State of Parana, Brazil, in 2015, and received the Ph.D. degree in Pontifícia Universidade Católica do Paraná, she is a full professor at the Graduate Program on Computer Science in University of Unicentro since 2006. Her research interests are mobile computing, distributed algorithms and data.
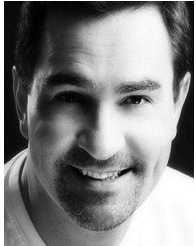
Luiz A. P. Lima Jr. is a full professor at the Graduate Program on Computer Science-Pontifícia Universidade Católica do Paraná, Brazil. He has got his Ph.D. degree from the National Institute of Telecommunications (INT/Université d'Évry), France and his research interests include middleware, mobile computing, distributed and parallel algorithms and simulation frameworks.

Gil Eduardo de Andrade received degree in computer engineering in 2005 from Pontifícia Universidade Católica do Paraná, Brazil, is a full professor in Instituto Federal do Paraná since 2010 and his research interests include mobile computing, distributed and parallel algorithms.

Aélio de Oliveira Jr. is a full professor at the Department of Information Systems -Federal University of Sergipe. He has got his Ph.D. degree from the Pontifícia Universidade Católica do Paraná, Brazil and his research interests include delay tolerant networks, digital TV, mobile computing and distributed and parallel algorithms.