

Service Evolution in a Nomadic Wireless Environment

Iacopo Carreras, Francesco De Pellegrini,
Daniele Miorandi, and Hagen Woesner

CREATE-NET,
via Solteri 38,
Trento – 38100, Italy
`name.surname@create-net.it`

Abstract. In this paper, we present and analyze a framework for self-evolving autonomic services in a wireless nomadic environment. We present a disconnected network architecture, where users mobility is exploited to achieve a scalable behaviour, and communication is based on localized peer-to-peer interactions among neighboring nodes. Service management is achieved by introducing autonomic services, whose operations are based on a distributed evolution process. The latter relies on the concept of *mating*, i.e., the exchange of information (e.g., code, parameters, data) among service users, which collaborate to enhance their *fitness*, defined as the ability of the actual service to fulfill the environmental features. The core of the evolution process is given by the service mating policy, which defines the way the running services should be modified when mating with other users. We introduce a general framework for analyzing service mating policies and exploit results from martingales theory to study their convergence properties. In particular, we introduce two optimal policies, clone-and-mutate and combine-and-mutate, and analyze their convergence times through extensive numerical simulations, addressing the impact of various parameters (number of nodes, users speed, mobility pattern).

1 Introduction

The emerging of a novel pervasive computing environment imposes big challenges to the current ICT technologies, calling for novel paradigms of communication, computing and service provisioning. The Internet, as we know it, is going to explode, due to the tremendous amount of information exchanged among a massively large number of devices. This problem does not concern only the communication aspect, i.e., the ability of the network to carry information, but also the way in which the network is managed and administrated. New paradigms are needed to face the needs and features of this novel networking environment.

In particular, autonomic communication systems are expected to represent one of the major technological breakthrough in the next decades, enabling the

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: [10.1007/978-3-540-32993-0_29](https://doi.org/10.1007/978-3-540-32993-0_29)

I. Stavrakakis and M. Smirnov (Eds.): WAC 2005, LNCS 3854, pp. 29–40, 2006.
© Springer-Verlag Berlin Heidelberg 2006

introduction of novel services and leading to a deep change in people/technology interactions [1]. The term “autonomic” comes from the computing field [2], where it is used to define systems which are inherently self-configuring, self-optimizing, self-healing and self-protecting. As the term itself suggests, an autonomic system should show the same properties of the human nervous system, which controls in an “unconscious” way routine tasks such as blood pressure, hormone levels, heart and breathing rate. Moving to the communication field, a system able to exhibit such features would clearly allow to solve the scalability and management issues related to the deployment of large-scale networks. In particular, we are interested in (i) defining a network architecture able to scale well with the number of nodes (ii) design services that are able to exploit the peculiar features of such networking support in order to evolve, leading, on the whole, to a complete autonomic communication system.

In this work, we first review, along the lines of some previous work by the same authors, a nomadic wireless network infrastructure model [3], which achieves network scalability by exploiting the users mobility [4] and a particular form of information filtering, that spatially limits the diffusion of data in the network [5]. Then, we address the problem of obtaining autonomic services over such a backbone-less networking infrastructure. We will see how such a task can be accomplished borrowing notions and drawing inspiration from the biological concept of evolution [6]. The basic idea is to have a service able to adapt to the surrounding environment, where the environment can include the space/time location of the user, the state of the entire system, the user’s requirements etc. The problem becomes then how to achieve service evolution over our infrastructure-less network. The solution we propose is based on a one-to-one mapping from biology to networking, in which a *population* (i.e., the instances of the service running on different users) evolves through *mating* (i.e., the exchange of code/parameters between neighboring nodes made possible by the adoption of a wireless interface), the mating process being driven by *fitness* (i.e., the ability of the service to fit the actual environmental features). In this way, a distributed version of the “survival of the fittest” paradigm can be applied to achieve adaptation by evolution. The result is a service which is able to evolve, without the need of any human intervention nor of a central controller, to adapt to the actual features/needs/requirements. The aim of this paper is to study this evolution process, first introducing a rather general model for the evolution and convergence of the service fitness level, and then addressing, through numerical simulations, the performance of this evolution process, covering a wide range of system parameters. In this way, we are able to obtain some insight into both the performance obtainable by self-evolving services, as well as into some desirable features of the actual evolutionary algorithm to be employed.

The remainder of the paper is organized as follows. Section 2 defines the networking setting we will focus on and introduces concepts and features of the self-evolving services we will consider. Section 3 presents a mathematical model for the fitness evolution process, analyzes three simple evolutionary mating algorithms and presents numerical results on the convergence process. Section 4

concludes the paper with a brief summary of the results obtained and points out directions for future investigations.

2 Autonomic Networks and Services

2.1 A Nomadic Wireless Networking Infrastructure

We consider a scenario where the nodes of a wireless network are attached or otherwise assigned to a human user. They are the central part of the electronic “halo” that will surround the human being in the future, making life easier, more enjoyable and more secure. It is reasonable to assume that the majority of nodes in this halo will fulfill only primitive tasks like sensing the environment and will not be able to perform complex computing as it may be necessary for the rendering of 3D-graphics or database operations. A small number of complex multi-purpose devices will then be controlling the primitive embedded systems, gather and process their information and provide an interface to the user. This separation in the functionality of the nodes follows the technological constraints imposed by the size (and corresponding energy/storage capacity) of the respective devices. In fact, many of those tiny devices will be in part or totally passive, relying on the RFID principle. Consequently only the more complex nodes (named user-nodes in [3]) will be able to exchange information among each other. The nodes of this network are mobile because users are mobile. In addition, the information that is being processed will be of local significance most of the time, which calls for a novel communication paradigm based on localized interactions among peer nodes rather than the conventional end-to-end approach. The access to a fixed Internet backbone is optional and will not be present all the time. If needed to provide a certain service, the user node is responsible for setting up the backbone connection. In addition, the kind of information that is going to be exchanged as well as the addressing (node ID vs. geographic address) will depend on the service running on the user node.

All these considerations lead to a network model where we do not assume packet relaying between the nodes. All communication will take place in a single-hop broadcast. In result we see a fully distributed network of mobile wireless nodes without any backbone connectivity. Routing (or the rules of information exchange) will be part of the service. We are not anymore considering packet forwarding, but instead *information* flow between nodes of the network. Services are self-contained, i.e. they function out of themselves and get information through interaction with the environment. Two basic problems arise: The first one is that of a network-wide information exchange using only single-hop broadcast and the second problem is what we call the management problem. While the authors showed at least the solvability of the first problem in previous publications (using information filters inside example services) this paper is intended to lay the ground to solve the second problem.

Management of a network usually involves different aspects like performance, configuration, accounting, fault, and security management. In traditional networks these tasks are performed in a centralized network management system (NMS). With the network model described above this approach is no longer feasible, and new concepts are needed.

2.2 Self-evolving Autonomic Services

If we target the same level of efficiency of current centralized network management systems, the scenario outlined above represents an harsh environment for two key reasons. First, the nomadic network paradigm is designed to deploy services on large-scale networks [3]. Clearly, centralized management is out of question due to scalability problems with respect to the number of nodes and active services. Second, a large fraction of the information conveyed in such a network will have local scope, as in the case of the estimation of certain physical quantities (temperature, pressure) or activity detection on the battlefield, so that a distributed solution sounds more promising. Furthermore, since the nomadic wireless networking paradigm we propose is based on disconnected operations, this clearly clashes with the needs of a centralized controller. We believe that the adoption of evolving services, i.e., services that are able to configure, regenerate and optimize their behavior, is the natural solution to such challenges. In this way, no centralized management mechanism is needed, and we only need to design supporting mechanisms for evolving and adapting services. Basically, the gain is that the network management is embedded in services and thus, since in the autonomic scenario services are users-situated, this calls for distributed operations and self-adaptation capabilities. Notice that, together with the end-to-end communication paradigm, we drop also the client-server semantics which is at the basis of the current Internet, replacing it with a novel paradigm, where services are carried on users' devices and evolve exploiting local peer-to-peer communications. In the rest of this work we will provide insight into some viable mechanisms to enable self-adaptation of autonomic services over the nomadic wireless infrastructure. Following some preliminary work by the same authors [6], our proposed approach is based on some biologically-inspired techniques. In particular, we focus on evolutionary paradigms for services: services should evolve showing the ability to drift towards better performance, resembling what several biological entities do. The key mechanism is the exchange, through a mating procedure, of data/code/parameters with other users. In other words, we exploit nodes mobility to enable node cooperation, shaped as exchange of code and/or parameters, so that the overall effect is a *distributed evolution process*. The success of this evolution process is quantified through a standard metric. i.e. the *fitness*, which represents also the driving parameter of the mating process. In the next section we shall illustrate that it is possible to provide an abstraction of the concept of service evolution and to obtain, analytically and numerically, results on the performance of the distributed evolution process.

3 On the Fitness Evolution

3.1 The Fitness Convergence Process

According to the framework outlined in the previous section, we are interested in modelling the evolution process of autonomic services over a nomadic wireless network architecture. It is worth remarking that, from the user's point of view, the process of service evolution is completely transparent, since what it experiences, in reality, is the evolution of the degree of satisfaction to the actual service provided, i.e., what we call *fitness*. In order to keep the analysis and simulation scenarios simple, we will employ a simple yet general model for the service and its associated fitness level. This sort of black-box approach to services comes from the observation that the evolution of fitness as an outcome of the evolution of the service code is something which should be evaluated on a case-by-case basis, whereas we are interested in getting insight into a more general framework. This is expected to provide useful information for the design of service mating policies, i.e., the algorithms that will actually drive the service evolution. In the following, we will provide, as examples, three possible service mating policies, derive the associated fitness mating functions (i.e., the functions that defines the fitness evolution process) and study their convergence properties. While this is not meant to be omnicomprehensive, it enables us to individuate stable and optimal service mating policies and to gain insight into the various factors influencing the design of an effective service mating algorithm. In particular, we are interested in understanding how some factors (i.e., the number of nodes, the nodes speed, the mobility model) affect the evolution process. In terms of fitness, we expect that services with a higher degree of fitness will have a higher chance to survive, so that, in the mating process, their *genes* (e.g., routines, code parameters etc.) are likely to be inherited by the offsprings.

We denote the fitness level of user i at time t as $I_i(t)$, and assume that $0 \leq I_i(t) \leq 1 \forall i = 1, \dots, N$, where N is the total number of users in the network. We can group the fitness level of all users into an N -dimensional vector $\hat{\mathbf{I}}(t) = [I_1(t), \dots, I_N(t)]$. Assuming that the users requirements and the environmental features are slowly changing over both time and space dimensions, $\hat{\mathbf{I}}(t)$ will change only at the mating instants. If the mobile speed is finite, the mating instants form a sequence $\{t_k\}_{k \in \mathbb{N}}$. In general, $\hat{\mathbf{I}}(t)$ will then be a random jump process defined on a suitable probability space $\{\Omega, \mathcal{F}, \mathbb{P}\}$. We denote by $\mathbb{E}[\cdot]$ the expectation taken with respect to the measure induced by \mathbb{P} . By standard arguments, we can transform $\hat{\mathbf{I}}(t)$ into a right-continuous left-limited (*càdlàg*) process, that will be denoted by $\underline{\mathbf{I}}(t)$. In order to study the system evolution, we can then limit our scope to the embedded process $\underline{\mathbf{I}}(t_k)$, where t_k denotes the k -th mating time. Please note that the mating times will be defined as a subset of the meeting times (i.e., the time instants two or more nodes get into mutual communication range), depending on the actual service mating policy employed (see below for the definition of the three cases considered). Also, note that the mobility models employed play a crucial role, in that they determine

the intensity (and distribution) of the sequence of meeting times, and thus the convergence rate of the service evolution process.

We consider the following two random processes:

$$X(t) = \frac{1}{N} \sum_{i=1}^N I_i(t); \quad (1)$$

$$Y(t) = \min_{i=1, \dots, N} I_i(t). \quad (2)$$

An easy sample-path argument leads to the following:

Lemma 1. *$X(t)$ converges to 1 \mathbb{P} -almost surely if and only if $Y(t)$ converges to 1 \mathbb{P} -almost surely.*

We then define the following:

Definition 1. *A service mating policy is called stable if it leads to convergence of $X(t)$ [$Y(t)$] with unitary probability.*

Definition 2. *A service mating policy is called optimal if it leads to convergence of $X(t)$ [$Y(t)$] to 1 with unitary probability.*

Please note that the condition of optimality is, in general, not sufficient for a mating policy to be efficient. How it will be discussed in the next section, we actually want a service that is able to converge fast to the optimal operating point, which actually regards the features of the transient behavior of the process $X(\cdot)$, while optimality here refers to a steady-state characteristic. Next, we want to establish some general sufficient conditions ensuring convergence of $X(t)$ [$Y(t)$]. We recall from [7] the followings:

Definition 3. *A process Z_n is said to be a submartingale (with respect to its natural filtration) if, $\forall n$, $\mathbb{E}[Z_{n+1}|Z_0, \dots, Z_n] \geq Z_n$.*

Theorem 1. *Let Z_n be a submartingale such that $\sup(|Z_n|) < +\infty$. Then there exist a random variable Z_∞ such that $Z_n \rightarrow Z_\infty$ with $\overset{n}{n}$ unitary probability.*

Clearly, it would be desirable to have a service mating policy which leads to $X(\cdot)$ be a submartingale, so that we have convergence of the evolution process. Further, the optimal convergence should be to a random variable with unitary mass at 1, so that the process of evolution will reach the optimal system operating point. We assume that the service of user i can be represented as a binary vector $\underline{v}_i = [v_i(1), \dots, v_i(T)]$, $v_i(l) \in \{0, 1\}$, $l = 1, \dots, T$.¹ The fitness is then taken to be

$$I_i = \frac{\sum_{l=1}^T v_i(l)}{T}.$$

¹ The representation of the service as a binary string is fully general, in that it applies to any ICT service. This abstraction, while enabling a general tractation, results in a simplistic approach with respect to “real” services. This complies with the main focus of the paper, which is to gain a deep understanding of this distributed evolutionary process; the application of the proposed framework to actual services is not straightforward, and is left for future work.

We introduce the following service mating policies:

Definition 4 (Clonation mating policy). *Let us consider two nodes with respective fitness level I_1 and I_2 that get into mutual communication range. Let us assume, without any loss of generality, $I_1 > I_2$ (if $I_1 = I_2$ no mating takes place). Then user 2 downloads (clones) user 1's service. User 1 keeps its service unchanged.*

Definition 5 (Clone-and-mutate mating policy). *Let us consider two nodes with respective fitness level I_1 and I_2 that get into mutual communication range. Let us assume, without any loss of generality, $I_1 \geq I_2$ (if $I_1 = I_2 = 1$ no mating takes place). If $I_1 = I_2$, we assume node 2 to perform the mating. Then user 2 downloads user 1's service. Mutation is then performed on the new vector \underline{v}_2 , by changing each digit independently with a given probability p (called the mutation probability). User 1 keeps its service unchanged.*

Definition 6 (Combine-and-mutate mating policy). *Let us consider two nodes with respective fitness level I_1 and I_2 that get into mutual communication range. Let us assume, without any loss of generality, $I_1 \geq I_2$ (if $I_1 = I_2 = 1$ no mating takes place). If $I_1 = I_2$, we assume node 2 to perform the mating. User 2 downloads user 1's service, i.e., the vector \underline{v}_1 . A number k is uniformly taken in the set $\{1, \dots, T\}$. Then, a new vector $\underline{v}'_2 = [v_1(1), \dots, v_1(k), v_2(k+1), \dots, v_2(T)]$ is formed. Mutation is performed on this vector, by changing each digit independently with probability p (called the mutation probability). User 1 keeps its service unchanged.*

To illustrate the fitness evolution process associated with such policies, let us consider the situation when two nodes, running the same service (but with different parameters), presenting fitness levels I_1 and I_2 , respectively, meet at time t_{k+1} . We assume, without any loss of generality, that $I_1(t_k) \geq I_2(t_k)$ and $I_2(t_k) < 1$. Both I_1 and I_2 are taken to be in the interval $[0, 1]$. In the case of $I_1 = I_2$, we assume without any loss of generality node 2 to perform the mating.

In general, we have $(I_1(t_{k+1}), I_2(t_{k+1})) = \phi[I_1(t_k), I_2(t_k)]$, where $\phi[\cdot]$ is what we call the *fitness mating function*, that maps $[0, 1] \times [0, 1]$ into itself. The mating function is, in general, taken to be a stochastic function, defined on $\{\Omega, \mathcal{F}, \mathbb{P}\}$. For the three examples considered above, the fitness mating function takes the following form:

$$\phi[x, y] = (x, x), \text{ clonation mating policy}, \quad (3)$$

$$\phi[x, y] = (x, x + \xi), \text{ clone-and-mutate mating policy}, \quad (4)$$

$$\phi[x, y] = (x, \psi \cdot x + (1 - \psi) \cdot y + \xi), \text{ combine-and-mutate mating policy}, \quad (5)$$

where ψ accounts for the combination operator and ξ is a random variable accounting for the mutation operator. From the structure outlined in the definition, it is clear that $\mathbb{E}[\xi] = 0$ and $\mathbb{E}[\psi] = \frac{1}{2}$.

We are interested in studying the convergence properties of the aforementioned policies. We assume that the initial fitness values are independently taken

from a continuous distribution $F_0(\cdot)$. We get the following results, whose proofs can be found in [8]:

Proposition 1. *The clonation/clone-and-mutate/combine-and-mutate mating policies are stable.*

Proposition 2. *The clonation mating policy is not optimal.*

Proposition 3. *The clone-and-mutate mating policy is optimal.*

Proposition 4. *The combine-and-mutate mating policy is optimal.*

3.2 Simulation Scenario

In order to study the effectiveness of the distributed evolution process, we run a wide set of simulations using a freely available software tools [9]. We simulated, for the whole range of simulation parameters, the two optimal service mating policies outlined above, and took the mutation parameter p equal to 0.1. We denote by N the total number of nodes, and assume that they are constrained to move in a square of $2000 \times 2000 m^2$. Each node is assumed to have a transmission range of 50 m, and IEEE 802.11b-compliant PHY and MAC protocols are used [10]. The nodes are initially dropped according to a uniform distribution on the square, and then start moving according to either a Brownian Motion (BM) or a Random Waypoint Mobility (RWM) at constant speed v taken in the set $\{2, 5, 10, 15\}$ m/s. For the RWM model, the pause time has been set to zero; also since we are interested in the transient behavior of the fitness level, the speed decay phenomenon does not play a significant role in our scenario [11]. Please note that our simulation of RWM is *not* a perfect simulation [12]. Indeed, we do not start the simulation with nodes distributed according to the steady-state distribution, but, rather, with a uniform one. This does indeed represent a pessimistic assumption, in that, as it may be easily understood, a uniform distribution of nodes over the area of interest is the distribution yielding the lowest probability of having nodes connected to one another (or, alternatively, the highest node isolation probability). Nonetheless, we believe that such an assumption leads to meaningful results in terms of comparison of the performance obtainable with BM and RWM. For the BM model, a billiard-like reflection was used when the mobile reached the edge of the domain. The initial fitness values are drawn from a set of i.i.d. random variables having uniform distribution in the interval $[0, 1]$. What we are interested in measuring is the *convergence time*, that will be defined in two ways. First, setting a threshold ξ (in the simulation, we will use $\xi = 0.95$), we want to measure the time it takes for the average fitness level to exceed ξ . Formally, $T_{conv}^{avg} = \min \left(t : \frac{\sum_{i=1}^N I_i(t)}{N} \geq \xi \right)$. Then, we are interested in the time it takes for the all the fitness values to exceed ξ , i.e., $T_{conv}^{min} = \min \left(t : \min_{i=1, \dots, N} (I_i(t)) \geq \xi \right)$. Clearly, $T_{conv}^{min} \geq T_{conv}^{avg}$. Further, the

smaller such convergence times, the more efficient the evolution process and the ability of the service to adapt to rapidly changing environmental conditions. Indeed, while the framework outlined in the previous section was able to answer our questions regarding the stability (i.e., the steady-state) of the distributed evolution process, we did not get any quantitative result on the convergence time, that is what in reality impacts the user's perception of the service quality.

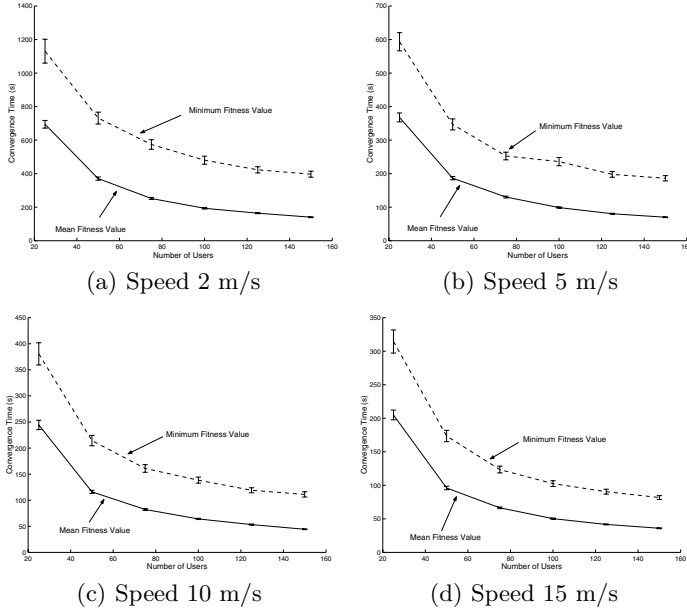


Fig. 1. Convergence times for the clone-and-mutate mating policy under the random waypoint mobility model

The first issue we want to address is which of the two optimal mating policies described in the previous section is able to achieve the faster convergence. The results, in terms of 95% confidence interval for the convergence times, for the RWM mobility model are reported in Fig. 1 and Fig. 2 for the clone-and-mutate and combine-and-mutate policies, respectively. It can be seen that the clone-and-mutate has, in general, quite lower convergence times, showing thus higher ability to adapt to changing environmental conditions. On the other hand, the combine-and-mutate is able to achieve interesting performance figures when dealing with high-density high-mobility scenarios, the most interesting cases for the pervasive environments we are targeting. The combine-and-mutate policy shows then to represent an interesting choice, and we are currently investigating whether more complex extensions of such a scheme can be actually used to speed up the service evolution process.

We also tested extensively the case of BM mobility model, which resulted in worse performance with respect to the RWM case. In Fig. 3 and Fig. 4 we

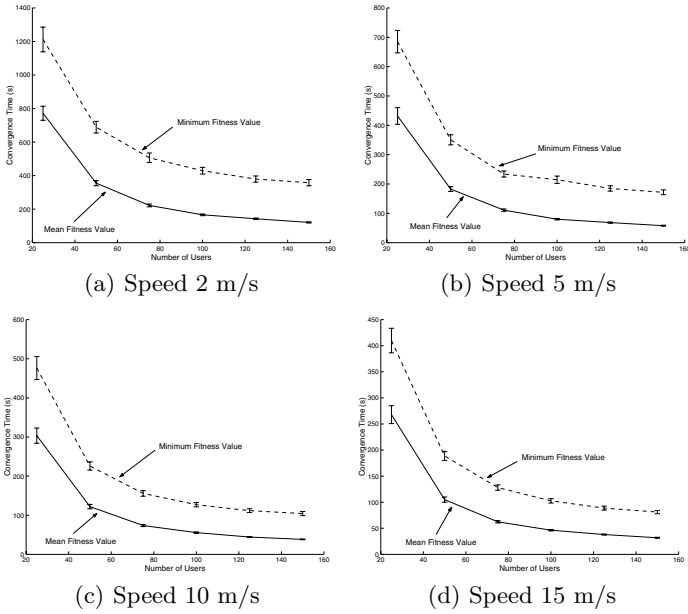


Fig. 2. Convergence times for the combine-and-mutate mating policy under the random waypoint mobility model

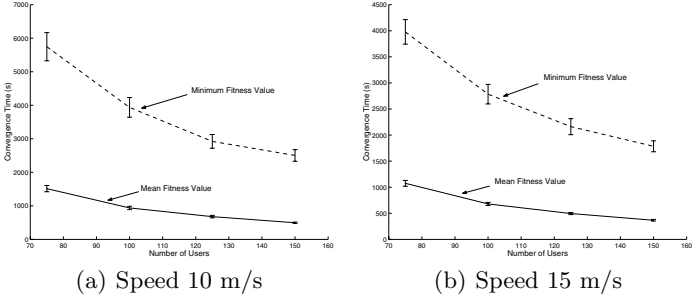


Fig. 3. Convergence times for the clone-and-mutate mating policy under the Brownian motion mobility model

reported the convergence times for the clone-and-mutate and combine-and-mutate policies. The results are reported, in terms of 95% confidence interval, only for the cases of $N = 75, 100, 125, 150$ users moving at a speed $v = 10, 15$ m/s, because of the extremely long convergence times under such a mobility model. As it may be easily seen comparing these results with the ones in Fig. 1 and Fig. 2, the RWM model is able to achieve much better performance (almost one order of magnitude), in terms of convergence time. This phenomenon is worth some comments. In general, it reflects the fact that the inter-meeting times in the RWM are smaller than in the BM (see [13] for an extensive and

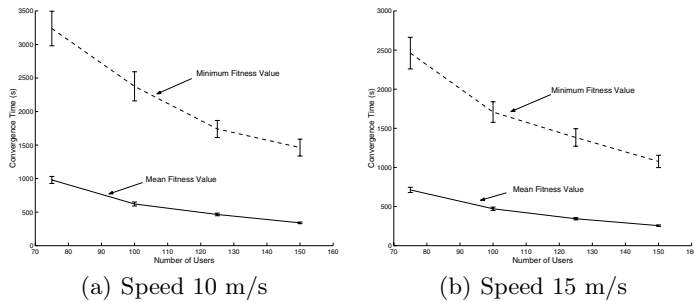


Fig. 4. Convergence times for the combine-and-mutate mating policy under the Brownian motion mobility model

in-depth discussion of such phenomena). However, the difference in the intermeeting times statistics is not sufficient to completely understand the difference between the two. Indeed, an extensive analysis of the trace files shows that the convergence (in particular for the minimum value) is driven by a few nodes which keep isolated for a very long time. This comes from the well-known tendency of the BM model to “move around”, without getting far from the initial location. Hence, if a node is very far from all the others in the initial distribution, it will take him a very long time before getting in contact with the rest of the population. Further, nodes will tend to remain in closed clusters; the evolution process remains localized inside the cluster and hence becomes much lower, since such a “local” evolution takes place over a smaller population. On the other hand, in RWM, nodes tend to pass through the center, and to meet more regularly with each other. In particular, a careful analysis of the trace files show that the time it takes for a node to get in touch with *all* other nodes is much lower in RWM than in the BM model. We might thus conclude that regularity in the traffic pattern helps in speeding up the convergence process.

4 Conclusions

In this work, we have proposed a framework for self-evolving autonomic services in a nomadic wireless environment. Freely drawing inspiration from the living world, we outlined a one-to-one mapping from biology to services, and apply the concept of evolution by adaptation to obtain truly autonomic services, able to self-optimize and to self-adapt to changing environmental conditions. The core of this evolution process is represented by the service mating policy. We outlined a general framework for studying service mating policies, and exploited results from martingales theory to assess the convergence properties of the resulting evolution process. We considered three policies and showed that two of them, the clone-and-mutate and the combine-and-mutate ones, are actually optimal, in that they are able to reach with unitary probability the optimal operating point. These two optimal policies have been widely compared through extensive numerical simulations, leading to interesting conclusions in terms of performance

impact of parameters such as the number of nodes, the nodes speed and the mobility pattern.

In order to improve the present work, one direction of interest is to build a model for the transient behaviour of the evolution process, in order to get estimates (or, at least, bounds) for the convergence time of the fitness evolution process. Further, an open issue remains the characterization of the mating times sequence for different mobility models, in order to analyze the impact of the mobility pattern on the evolution convergence rate. Finally, we are currently investigating more complex extensions of the combine-and-mutate service mating policy, exploiting results in the area of GAs to find solutions able to speed up the convergence process in highly-dense highly-mobile networks.

References

1. M. Smirnov, "Autonomic communication: research agenda for a new communication paradigm," 2004, white Paper. [Online]. Available: http://www.autonomic-communication.org/publications/doc/WP_v02.pdf
2. J. O. Kephart and D. M. Chess, "The vision of autonomic computing," *IEEE Comp. Mag.*, vol. 36, no. 1, pp. 41–50, Jan. 2003.
3. I. Carreras, I. Chlamtac, H. Woesner, and H. Zhang, "Nomadic sensor networks," in *Proc. of EWSN*, Istanbul, Turkey, 2005.
4. M. Grossglauser and D. Tse, "Mobility increases the capacity of ad hoc wireless networks," *IEEE/ACM Trans. on Netw.*, vol. 10, no. 4, pp. 477–486, Aug. 2002.
5. I. Carreras, F. De Pellegrini, D. Miorandi and I. Chlamtac, "Information filtering in a 2-tier wireless sensor network," in *Proc. of Sensorfusion*, Budapest, HU, 2005.
6. I. Carreras, I. Chlamtac, H. Woesner, and C. Kiraly, "BIONETS: BIO-inspired NExt generaTion networkS," in *Proc. of WAC*, Berlin, DE, 2004.
7. D. Williams, *Probability with Martingales*. Cambridge, UK: Cambridge University Press, 1992.
8. I. Carreras, F. De Pellegrini, D. Miorandi and H. Woesner, "Service evolution in a nomadic wireless environment," CREATE-NET, Tech. Rep., 2005. [Online]. Available: <http://www.create-net.org/~dmiorandi>
9. AA.VV., "Omnet++." [Online]. Available: www.omnetpp.org
10. *Supplement to 802.11-1999, Wireless LAN MAC and PHY specifications: Higher Speed Physical Layer (PHY) extension in the 2.4 GHz band*, IEEE Std., Sep 1999.
11. J. Yoon, M. Liu, and B. Noble, "Random waypoint considered harmful," in *Proc. of IEEE INFOCOM*, San Francisco, CA, 2003.
12. J.-Y. Le Boudec and M. Vojnović, "Perfect simulation and stationarity of a class of mobility models," in *Proc. of IEEE INFOCOM*, Miami, FL, 2005.
13. R. Groenevelt, "Stochastic models in mobile ad hoc networks," Ph.D. dissertation, INRIA, 2005. [Online]. Available: <http://www-sop.inria.fr/maestro/personnel/Robin.Groenevelt/Publications/Thesis.pdf>