

A Biologically Inspired Service Architecture in Ubiquitous Computing Environments

Frank Chiang and Robin Braun

Faculty of Engineering, University of Technology Sydney,
Broadway, NSW 2007, Australia
frankj@eng.uts.edu.au

Abstract. This paper describes the design of a scalable bio-mimetic framework in the management domain of complex *Ubiquitous Service-Oriented Networks*. An autonomous network service management platform - SwarmingNet is proposed. In this SwarmingNet architecture, the required network service processes are implemented by a group of highly diverse and autonomic objects. These objects are called Tele-Service Solons as elements of TeleService Holons, analogue to individual insects as particles of the whole colony. A group of TSSs have the capabilities of fulfilling the complex tasks relating to service discovery and service activation. We simulate a service configuration process for Multimedia Messaging Service, and a performance comparison is made between the bio-agents scheme and normal multi-agents scheme.

1 Introduction

The operational management of Next Generation Network (NGN) services is expected to be autonomous, scalable, interoperable and adaptable to the diverse, large-scale, highly distributed, and dynamically ever-changing network environment in the future. The functional management for them are also desired to be as simple as possible from the perspective of both designs and implementations. Current network management infrastructure is struggling to cope with these challenges.

In contrast, social insects and biological organisms have developed relatively easy and efficient mechanisms to thrive in hostile, dynamic and uncertain environments after many years' evolution and natural selection. Hence, taking advantages of the synthesis on full-scaled biological societies is of vital importance in achieving autonomic management in the future Ubiquitous Service-Oriented Network (USON) [1], which will dynamically connect human beings and home/office electronic appliances via distributed devices (e.g., cell phones, notebooks, PDAs) and applications running on these devices to enable services at any time, any places without *constraints in quantity or frequency* [2].

The aim of this paper is to propose a bio-swarmling framework — *SwarmingNet* for network service managements. The biological platforms proposed by Suda [3] and Suzuki [4] are more emphasizing *evolutionary* behaviors of agents. The status of network services (mutation, clone, reproduction and replication) depends on

multi-agents' internal states. These platforms are particularly complicated in terms of rapid practical application. We distinguish our framework from theirs by applying TeleSolon hierarchy^[1] concepts into hierarchical network management system which is analogue to the *ecosystem* in nature and *colony* structure in ants. We apply stigmergic ant-foraging behaviors into the threshold-based self-organization algorithm. This is much easier to be applied into practical embedded systems with industries.

This paper is organized as follows: The design principles and our self-organized provisioning algorithm for *event-based* autonomic management architecture are mainly specified in section 2. The system-level architecture are included in section 2. Section 2.1 covers the definitions and theorem for the threshold-based self-organized algorithm. Service configuration process of MMS is simulated as an illustrative application, simulation results are shown in Section 3. Finally we conclude with performance comparisons and future work trends in section 4.

2 Design Principles

The incorporation of social insects paradigm into autonomic service configuration is believed to be the right solution to meet these requirements. This can be achieved by modelling networks as a distributed aggregation of self-organized autonomous TSS solons in our approach. This is similar to the social insects colony (networks) consisting of large amount of individual insect (solon).

2.1 Self-organized Service Provisioning and Algorithm

Future networks should not only provide basic connectivity but also intelligently and immediately enable on-demand services in *pervasive computing* environments at anywhere, anytime. Those services must be provisioned in a flexible and distributive way in highly dynamic runtime infrastructure. Thus, service deployment and management for devices in USON is extremely difficult since a provisioning infrastructure ought to cope with the high level of heterogeneity, degree of mobility, and also take into account limited device resources. In this context of self-organization, we describe the service provisioning as the ability *to create, to remove, to reproduce, to reconfigure* the instances of services at runtime. Moreover, the bio-mimetic agents running at particular network nodes, (1) *measure* the local demands for network services autonomically beyond other nodes; (2) *reconfigure/reproduce* local services when demands are detected; (3) *remove* the services when there are no demands of quantity.

Self-organization Algorithm — This section defines the algorithm^[2] to enable self-organized bio-networks. This algorithm also considers the scalable design principle, such as adaptivity and robustness in distributed environment; localized

¹ Due to the limitation of pages, details of THSs and TSSs can be referred to in the full version under requests.

² Details of this algorithm and proof can be referred to in full version of this paper.

decision making process based on neighborhood information. Hence, it is also a practical principle to consider only local customers who are not too far from the service center. The content-based event messaging system categorizes and stores the messages for different services into different space in information store (e.g., database). The following definitions as a whole define the problem domain in our service provisioning process. Our self-organization algorithm considers both the factors of time and space as follows:

Definition 1. *A dynamic threshold θ is configured for each requested service respectively; A parameter η is used to evaluate the keen degree for customers who require certain services; A parameter d represents Euclidean distance between available server of services and customers; The localized service zone ω is designated as $\omega(d) \leq 10\% \cdot D$, where D is the diameter value of the whole service area.*

Definition 2. *A parameter τ evaluates the intensity of digital pheromone which are placed along the traces to servers in dispersed area by previous bio-agents. The intensity of digital pheromone measures how easy the service is available in the particular server the path gets to. Moreover, τ is related to the Euclidean distance d .*

Lemma 1. *(Time) When customer requests are accumulated to the θ value, the service provisioning starts. This is a dynamic value varying in accordance with essence of specific services. For some reason, some services should be activated as soon as there is a demand for it. While some services can be activated only when there are enough number of requests; (Space) When the customer and services are both inside localized service zone, the service could be provisioned.*

Theorem 1. *The self-organized process for service provisioning is activated successfully iff Lemma 1 and Lemma 2 are satisfied simultaneously.*

2.2 System-Level Architecture

This architecture we propose here is partly depicted in Figure 1 of our paper [5]. We have to omit it due to page limitations. This systems-level architecture illustrates the combinatorial links among our three indiscerptible parts for autonomous service activation process: 1) Users; 2) Instrumentation support and measurement, monitoring; 3) Enhanced 4-layers TMF management model. Analogue to the biological society, we introduce the concept of *ecosystem* into the whole system which acts as the environment where agents create, live and die. We designate the *energy exchange* is the "currency" between ecosystem components (e.g. swarm agents) and eco-environment.

3 Simulation and Experimental Measurement

We choose the Multimedia Messaging Service (MMS) as an evaluation application. As for the system-level architecture, the *managed object* in this context is MM_Box (MultimediaMessaging_Box); The product components are 1)

Gold_MM_Box (capacity=1000MB), 2) Silver_MM_Box (capacity=100MB), 3) Bronze_MM_Box (capacity=10MB).

The events messages contain on-demanding service provisioning requests from clients, these messages include information about: 1) *creation and deletion of users' MM account in the product* or 2) *migration Multimedia Messaging (MM) account among the products – Gold, Silver and Bronze boxes*. The service provisioning results indicate bio-inspired network management paradigm and maintain SLA compliance as well as efficient transaction time. Digital pheromone evaluates the degree of difficulty in activating or migrating MM accounts which are stored in MM servers in this context (The large intensity of digital pheromone means MM_boxes are easier to be migrated from silver to gold, or from bronze to silver, etc.). The effectiveness of digital pheromone in MMS server configuration process in the framework has been tested. The service-configuration performance comparison between the bio-agents and normal agents are analyzed. Java classes are built on the hybrid modelling platform AnyLogic[®].

The experiment scenario is summarized here: The event messages with service requests from clients PC in our testbed will trigger the service configuration process whenever the service requests approach a service threshold θ_{ij} where i represents the service ID, j represents the clients' ID.

We argue this is an *autonomic* process instead of an *automatic* paradigm because the θ_{ij} value changes according to the requested service profiles and autonomic agents learn and decide the best threshold. Our adaptation strategies are not depending on a set of preconfigured rules like that in automatic system, on the contrary, the autonomy is achieved by goal setting and suggestion through learning and modification of the existing adaptation strategy.

The multiplication ω of digital pheromone intensity and customer keen index will be considered as an important index in an exponential formula (e.g., $\exp(\omega)$), which determines which MM account will be configured to activate in certain MM_servers. Service lifetime is calculated by multiplication of these two factors. Moreover, network vendor agents go through caché database for updated information being synchronized with our 4-layer structure which covers the *specification* files for products and services, and all the configuration files for resources (e.g., devices, equipment, etc.). Specifically, by taking into fact that service requests usually are provisioned by local servers, the factor d , an Euclidean distance measuring our virtual distance between service requests to MM_server in the coordinate plane. If there are multiple servers meeting the requirement simultaneously, we will randomly pick up one of them.

Figure 1 describes the overall simulation configurations in details. Figure 2 shows the parameters of 3 experimental scenarios which test the service provisioning. As shown in Figure 2 for Experiment 3, the number of MM_servers is decreased till 100 while other parameters remain the same as above.

3.1 Experimental Results

Based on the three experimental test scenarios described in the previous subsection, the performance comparison between bio-agents and normal multi-agents

Topology	<ul style="list-style-type: none"> □ 1000 virtual MM servers are uniformly distributed into an area [0, 280]
Event Parameters	<ul style="list-style-type: none"> □ Requests for on-demanding service are randomly generated by clients with fixed seed=1 over a time interval of 20 days □ Services lifetime are not permanent, they will deceased whenever there are no needs or termination willing from customers. {We give maximum_service_lifetime = 1 day}
Space/Time Dynamics	<ul style="list-style-type: none"> □ Transaction time for each service requests dynamically change according to $A = \text{Customer_Keen_Index}$ and $B = \text{Digital_Pheromone_Intensity}$; □ In order to simplify the simulation factors, we set $A=0.5$; $B = \gamma \times d$, ($\gamma=0.5$ or 0.6) where $d = \sqrt{(x-x_i)^2 + (y-y_i)^2}$, d represents the Euclidean Distance between one particular service request and one particular MM server; □ Maximum duration for any certain service provisioning $\lambda=0.4$ days □ Services configuration happen on those servers which are close to customers like in real world. We only provision the service distance $d \leq 30$

Fig. 1. Experiment Description

	Num_of_MM_Servers	Initial_Value_Provisioned_Servers	Digital_Pheromone_Intensity	Customer_Service_Keen_Index
Test 1	1000	0	0	0.5
Test 2	1000	0	0.6	0.5
Test 3	100	0	0.6	0.5

Fig. 2. Parameters for Experimental Test 1, 2 and 3

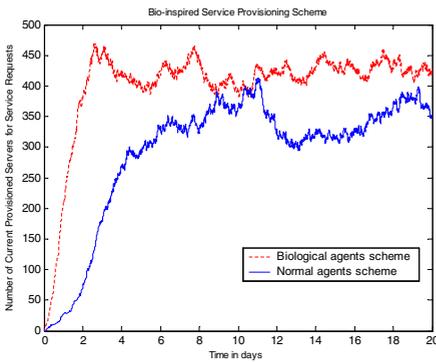


Fig. 3. Performance Comparison for Number of Servers Configured between Biological Inspired Agents Scheme and Normal Agents Scheme

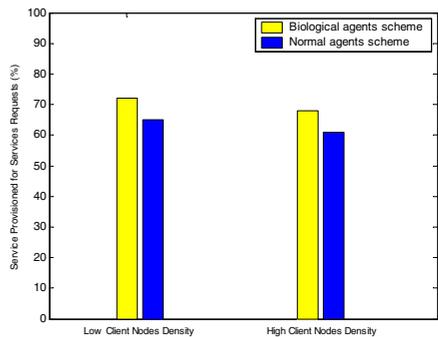


Fig. 4. Service Configuring Percentage in Heterogeneous and Dynamic Network Environment with Different Clients Density

without biological behaviors are presented in Figure 3. As we can see, the number of servers in service provisioning process is bigger for bio-inspired agent framework when the same quantity of service requests (=11692) arrives. Service configuration tasks, or workload are distributed uniformly into servers with a shorter response time, and performance for load balancing is also optimized by this bio-inspired framework.

We calculate the percentage of configured service of the total number of services, which though are finally provisioned respectively in the low client node density (=100 in the fixed area) and high client node density (=1000 in the fixed area). In low client requests environment, service configured percentage is 72%, which is higher than that in high client requests environment. However, our biological agents scheme results in better service configured percentages in the same environment as those normal multi-agent schemes. Details are illustrated in Figure 4.

4 Conclusion and Future Work

Firstly, we conclude that our bio-inspired multi-agent framework provides a solution to envision future *autonomic* service management system. This framework outperforms the current normal multi-agent based system in terms of service discovery and service assurance for future IP networks. Secondly, this framework does not rely on particular types of insects societies or colonies, i.e. agents could be entities in USON ranging from any hardware devices to robotic agents, or biologically-inspired software elements. Finally, our future work will specifically focus on the performance comparison among Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Genetic Algorithm (GA) with regards to efficient service configuration issues on the basis of this framework.

References

1. Yamazaki, K.: Research directions for ubiquitous services. In: Proceedings of International Symposium on Applications and the Internet, p. 12 (2004)
2. Suzuki, K.: Ubiquitous services and networking: monitoring the real world. In: Proceedings of ISAI, p. 11 (2004)
3. Nakano, T., Suda, T.: Self-organizing network services with evolutionary adaptation. *IEEE Transactions on Neural Networks* 16(5), 1269–1278 (2005)
4. Makinae, N., Suzuki, H., Minamoto, S.: Communication platform for service operations systems in advanced intelligent network. In: *IEEE International Conference on Communications*, vol. 2, pp. 872–877 (1997)
5. Chiang, F., Braun, R., Magrath, S., Markovits, S.: Autonomic service configuration in telecommunication mass with extended role-based Gaia and Jadex. In: *Proceedings of IEEE International Conference on Service Systems and Service Management*, vol. 2, pp. 1319–1324 (2005)