Crossroads
The ACM Student Magazine

Learning from Nature: Network Architecture Inspired by Biology

by***Anh Nguyen***,***Tadashi Nakano***, and***Tatsuya Suda***

Introduction

Mobile and peer-to-peer network services are rapidly growing to include a large number of users that frequently join and leave such networks [**4**]. These networks are often highly heterogeneous and that wide varieties of devices are networked with varying connectivity and availability. As these underlying networks increase in scale, dynamics, and heterogeneity, network services must address the issues of scalability (the ability to accommodate enormous numbers of users in large-scale networks), adaptability (the ability to handle the dynamics and heterogeneity of network conditions), and availability (the ability to provide fault tolerant network services).

The Bio-Networking Architecture is a new framework for developing network services that can satisfactorily meet such key requirements (scalability, adaptability, and availability) [**5**]. It is inspired by the observation of large-scale biological systems such as bee colonies: it applies key biological principles and mechanisms to the design of network services. Key characteristics and features of the Bio-Networking Architecture are the following.

- **Biologically-inspired behavior**: Network services are provided by autonomous agents called cyber-entities that implement rudimentary biological behaviors such as replication, migration, or death. They obtain a resource called energy in exchange for providing a service to users (or other cyber-entities), and consume it to survive in their network environment. This is analogous to real biological entities that acquire and consume energy (or food) for their survival. Also, they may interact with their network environment and other cyber-entities through environmental sensing and communication.
- **Self-organization** (or decentralization of control): Cyber-entities do not rely on centralized control. They are designed to act autonomously based on their internal states (e.g., age) and/or local environmental conditions (e.g., resource availability of their network environment). Network services created (i.e., collections of cyber-entities) are highly scalable since no global computation and coordination are required.
- **Evolution and adaptation**: Cyber-entities evolve their behavior through natural selection that occurs due to a difference in survival and reproductive success. For example, cyber-entities who successfully provide network services accumulate energy and grow into larger population. Evolution allows network services to become more adapted to their network environment as well as adapting to the changes in their network environment.
- **Emergent behavior**: Collective behavior of cyber-entities or interactions between cyber-entities may produce advantageous behavior (e.g., division of labor) that cannot be achieved by a single individual cyber-entity. Such behaviors often emerge as a result of evolution and adaptation.

In the following, the design principles and key features of the Bio-Networking Architecture are described in more detail.

The Bio-Networking Architecture

Design Principles

In biological systems, beneficial features often emerge through simple interactions in a group of individuals. For example, an insect society such as an ant colony acts as a superorganism capable of efficiently exploiting food sources through the simple interactions of individual ants.

The concept of emergent behavior is applied to the Bio-Networking Architecture by

modeling a network service as a group of autonomous interacting agents called cyber-entities. Individual cyber-entities (e.g., ants) can be simply designed, yet a group of cyber-entities (e.g., an ant colony) is expected to exhibit useful properties such as scalability and adaptability.

Cyber-entities implement a specific network service (e.g., a content hosting service, a caching service, or a query processing service) and provide the service to their users (i.e., service consumers or other cyber-entities). They also implement autonomous actions such as replication, migration, and death. For example, they may replicate when they perceive increased demands for their network service; others may migrate toward users located far away; some others may die if their service become unpopular.

Cyber-entities reside on network platforms. Such network platforms provide cyber-entities with network and computing resources including network bandwidth, processing power, memory, and storage space. Cyber-entities use these resources to provide service, perform replication (make a copy of themselves), and/or migrate across platforms.

Evolution and Adaptation

Natural selection or the survival of the fittest is known to be a driving force behind biological evolution and adaptation. In the Bio-Networking Architecture, energy is introduced for natural selection to occur. Energy is given to cyber-entities in return for providing a service to users, and cyber-entities with abundant energy are allowed to replicate. On the other hand, cyber-entities consume energy when they use network and computing resources on platforms (e.g., CPU, memory, bandwidth, etc.). If cyber-entities are unable to pay energy for resources that they use, they are eliminated by platforms.

The idea of using energy allows all cyber-entities to survive as long as they retain energy. Diverse cyber-entities may be able to survive, which makes a network service collectively more robust against environmental changes in networks. In addition, natural selection occurs in a decentralized and localized manner, thereby a network service becomes more scalable since it can avoid a single point of failure. The localized nature of natural selection also allows a network service to become more adapted to specific network conditions.

Evolution by natural selection requires diversity or variability among cyber-entities as

is necessary for biological evolution to occur. When a cyber-entity population contains a low degree of variation, natural selection can not act, thus no evolution is achieved by natural selection. In the Bio-Networking Architecture, diversity is retained by means of mutation or random changes that modify a cyber-entity's behavior. Natural selection changes the frequencies of specific cyber-entities in their environment; the number of cyber-entities who are better able to replicate and survive increases, whereas the number of cyber-entities that are incapable of surviving or replicating decreases, eventually leading those cyber-entities to become extinct.

The natural selection just described differs from selection traditionally used in evolutionary computation (EC) such as Genetic Algorithms (GAs) [**2**]. In GAs, fitness functions are explicitly defined to select and replace solutions, whereas no fitness functions are used for natural selection, instead allowing all solutions (or cyber-entities) to survive as long as they retain energy. The end result expected would be increased diversity in a group of cyber-entities, making them collectively more robust against the changing network environment.

In addition to the natural selection described above, cyber-entities may perform preferential selection during reproduction [**3**]. A cyber-entity selects a partner and produces an offspring cyber-entity, where an explicit fitness function is used as in GAs. The selection can be performed based on some criteria. Suppose that selection is performed based on resource consumption of cyber-entities (i.e., cyber-entities that do not exhaust computing resources are more likely to be selected as a reproduction partner), then cyber-entities may be directed to evolve to conserve computing resources. Preferential selection performed by cyber-entities is still unlike the traditional GAs that perform selection in a centralized manner. Preferential selection in reproduction as well as natural selection in the Bio-Networking Architecture occurs locally allowing a network service to be more adaptive to the locality of network conditions.

Cyber-entity Behaviors

The following describes a set of behaviors that cyber-entities may possess. A more detailed description of cyber-entity behaviors is done in [**5**].

- **Energy acquisition and consumption**: Cyber-entities provide a network service to users or other cyber-entities and acquire energy. They expend the energy to use computing resources on networks.

- **Replication and reproduction**: Cyber-entities make copies of themselves (replication). Two cyber-entities may create a child cyber-entity (reproduction). Replication and reproduction may be performed when service demands increase. Cyber-entities may also do so to increase the availability of their service in case of network failures.
- **Migration**: Cyber-entities may migrate about networks. Various migration policies can be implemented. For example, cyber-entities may migrate randomly, they may migrate toward their users, or they may migrate to achieve a load-balance among hosting computers.
- **Hibernation**: Cyber-entities may go into a hibernation state when there is no demand for the service that they provide. Hibernating cyber-entities can conserve computing resources by reducing memory and CPU usages of the network computers, while they are ready to provide a service once activated.
- **Death**: Cyber-entities may die because of energy starvation. Also, they may voluntarily die when user demands for their service are low.
- **Communication**: Cyber-entities communicate with others for exchanging information that they have.
- **Environmental sensing**: Cyber-entities may sense the information of network and computing resources (e.g., CPU cycles and memory space) on neighboring platforms.

Simulation Study of the Bio-Networking Architecture

A set of simulations has been performed to study the inherent behavior of the Bio-Networking Architecture. Here, the simulation scenario considers a distributed service placement in a peer-to-peer network. Such a network is formed by a number of users, each of which has installed a Bio-Networking platform to host cyber-entities on his or her computer. Cyber-entities that provide a query processing service are running on the platforms and shared among those participating users. Users may collect energy by providing their computing resources to cyber-entities, and use the energy to request a network service that they need.

Distributed service placement in general may be defined as an optimization problem of deciding the number and location of services to place on a network so as to minimize certain performance criteria such as service time delay. Such a placement problem is often given with a resource availability constraint assuming that only limited amounts of resources are available on each platform; for example, each platform is able to host

only limited numbers of cyber-entities at a time.

Our approach toward the service placement is to use evolving cyber-entities to find out a solution in a distributed and competing manner. Resource constraints can be accommodated in such a way that each platform (i.e., a participating user) charges cyber-entities more energy when a resource constraint is being violated. Cyber-entities that are charged a large amount of energy eventually leave the platforms, releasing computing resources on the platform.

Simulation Model

The network is structured as a 10x10 network containing 100 platforms (**Figure 1**). In the network, randomly selected platforms generate 150 queries per second. A cyber-entity requires a service time of 1.0 seconds to process a query, and thus it takes one of the following two states: "busy" while processing a query, or "idle" otherwise. A user query generated is always forwarded to the nearest and idle cyber-entities. If all cyber-entities are busy, the query is placed on a waiting queue on a network platform. A cyber-entity that processes a query is rewarded with 10 energy units, while a cyber-entity consumes the cost of using platform resources amounting to 1 energy unit per second. Cyber-entities also consume 500 energy units when they perform replication and migration. Cyber-entities who run out of energy are deleted by platforms (i.e., through natural selection).

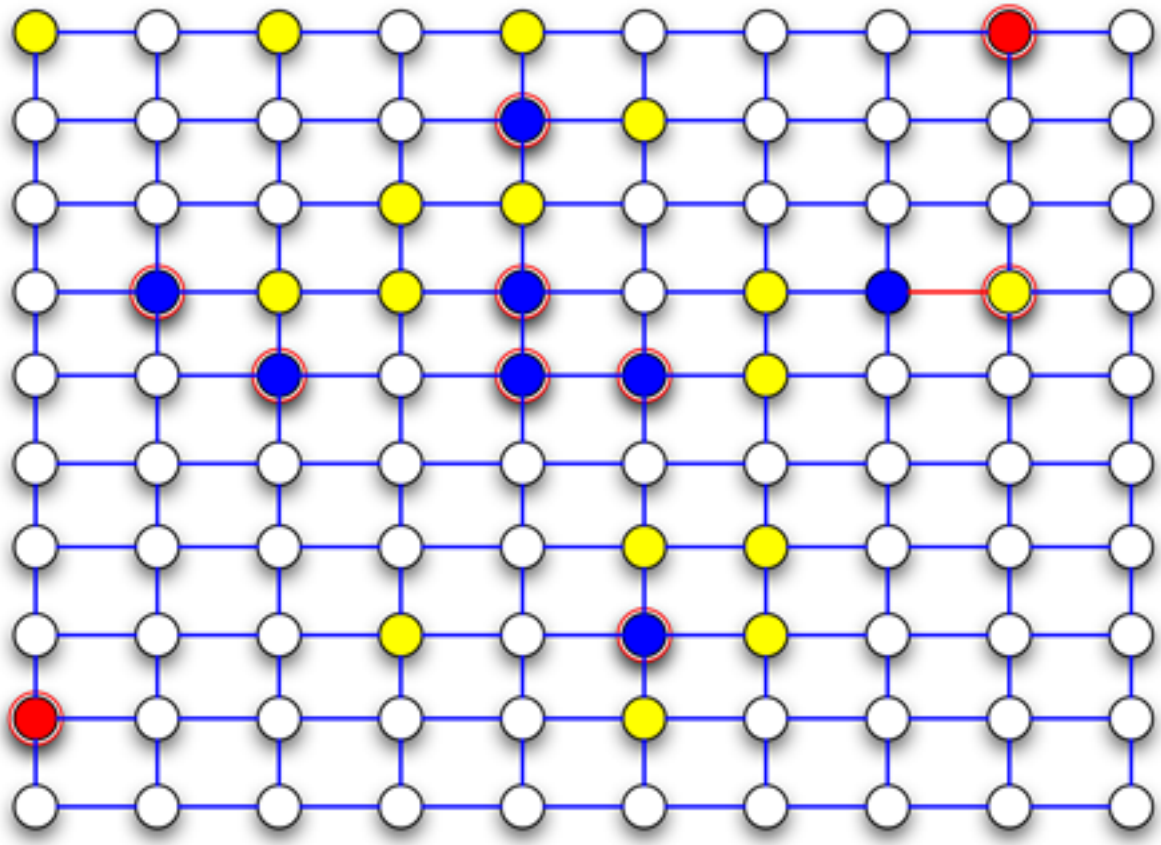

Figure 1: Simulated network using the Bio-Networking Simulator [**8**]. Network platforms are colored in terms of service demand (number of queries) and service supply (number of cyber-entities); white when there are no demands and no supplies; blue when all demands are satisfied; yellow when supplies outpace demands; red when demands exceed supplies.

Behavior Implementation

Cyber-entities make a decision on what behavior to perform given a set of network conditions and their internal states. The decision making algorithms can be implemented in various ways, for example, using neural networks [**7**] or genetic programming [**1**]. Cyber-entities to be simulated here implement two behaviors, replication and migration, whose decision is made based on weighted-sum aggregation. For example, replication behavior is implemented by a set of weights$w = \{w_1, w_2, w_3, \ldots, w_n\}$ and a set of sensory inputs from the environment and their internal conditions$i = \{i_1, i_2, i_3, \ldots, i_n\}$, where all elements of$w$ and i are represented as real values. Then, replication is invoked when$\sum w \cdot i > \theta_r$ (replication threshold).

Sensory input values can be perceived by cyber-entities and used in deciding whether

to perform a behavior. The following are some of the examples of information about themselves and their network that cyber-entities may be able to perceive.

- **Energy level**: the difference between acquired energy and consumed energy
- **Request rate**: the amount of incoming service requests on their network platform
- **Resource cost**: cost of computing resources on their network platform
- **Behavior cost**: cost of performing behavior (reproduction and migration) on their network platform
- **Population**: the number of cyber-entities on their network platform
- **Activeness**: willingness to invoke behavior

Weight values are determined when cyber-entities are born through replication. They are either inherited from parents, assigned, or randomly chosen values through mutation that occurs based on a probability called mutation probability. A large mutation probability produces diverse cyber-entities with different behavior patterns allowing for first evolution, which might be useful in a dynamically changing network but not in a stationary network (that does not require any change). A small mutation probability allows for a fine-grained tuning. The mutation probability used in the simulation adaptively varies depending on the extent of satisfaction of the network service that cyber-entities provide, which is measured by resource consumption and response time. The extent of satisfaction is locally observable by each network platform, and when either response time or resource consumption is not satisfactory, a mutation probability is raised. Thus, cyber-entities undergo changes in their behavior when their behavior is not suitable for their network environment.

In evaluating the performance of network services, the following two performance measures are concerned.

- **Resources consumed**: the amount of energy consumed by cyber-entities on network platforms
- **Response time**: the average time needed for a generated query to be processed by a cyber-entity

There is a trade-off between resources consumed and response time since as the number of cyber-entities increases, resource consumption increases, which should decrease the response time. The goal here is to evolve cyber-entities to balance those two performances. In other words, it is to decrease resource consumption while

keeping the response time low.

Simulation Results

Simulations initialize cyber-entities with randomly configured weights, and then place them on randomly selected platforms at the beginning of each simulation. **Figure 2** shows the average amount of resources (or energy) consumed by cyber-entities per simulation second (simulation cycle), and **Figure 3** shows the average response time perceived by users. Note these are the typical simulation results over a number of simulation runs.

Resource consumption is improved over time while response time is kept satisfactorily low. This is due to the evolution that has gradually adjusted cyber-entity behaviors (i. e., a set of weights). Although resources consumed and response time both sometimes fluctuate because of the random nature of mutation, it is demonstrated that cyber-entities have the ability to balance well the service demands (number of queries) and service supplies (number of cyber-entities) in a fully decentralized manner without using a central server.

The simulated network shown here is simple and more experiments in realistic networks are needed to demonstrate the scalability and adaptability of the Bio-Networking Architecture. A simulation study of the Bio-Networking Architecture has been going on [**3**], and the various aspects and detailed characteristics will be identified in the future through extensive sets of simulations.

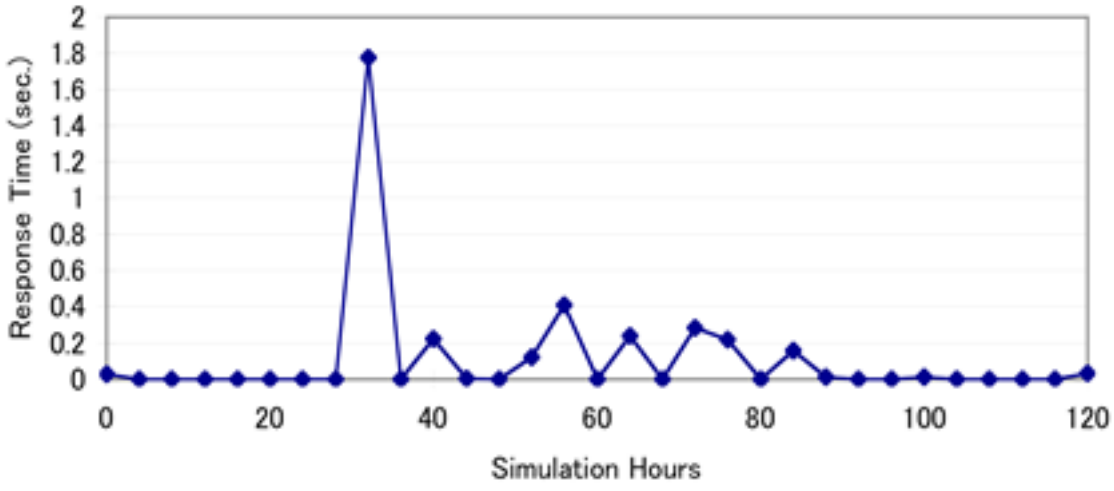

Figure 2: Resources consumed.

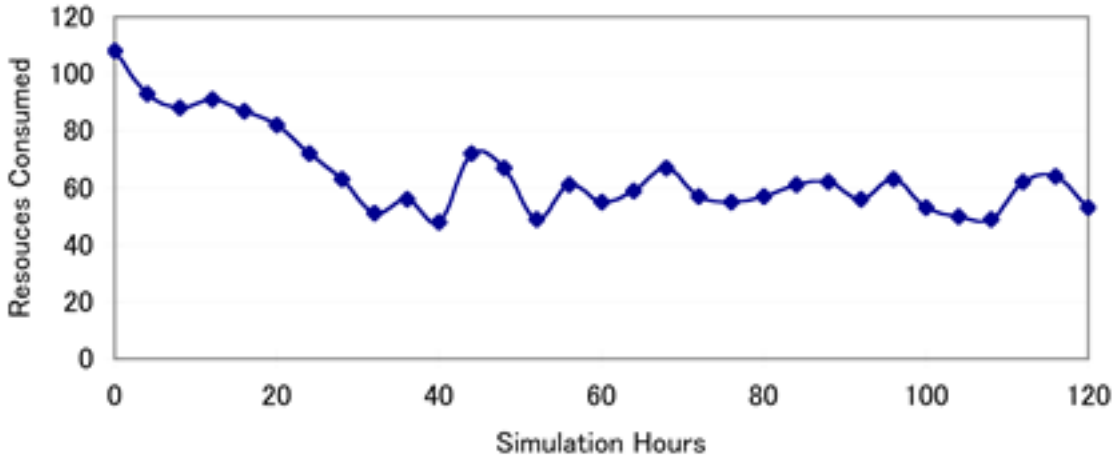

Figure 3: Response time.

Conclusions

This paper presents a biologically inspired approach for designing scalable and adaptive network services. Network services in the Bio-Networking Architecture should be scalable since cyber-entities do not rely on centralized control. Network services are also adaptive to various network conditions since cyber-entities have the ability to evolve. More information about the Bio-Networking Architecture is available at our project web site [**8**] where various softwares and documents to study the Bio-Networking Architecture can be downloaded.

References

1

Koza, J. (1992). Genetic programming. The MIT Press.

2

Mitchell, M. (1996). An introduction to genetic algorithms. The MIT Press.

3

Nakano, T., and T. Suda. (2004). Adaptive and evolvable network services. In *Proceeding of the Genetic and Evolutionary Computation Conference (GECCO-2004)* Lecture Notes in Computer Science, vol. 3102, Springer, pages 151-162.

4

Oram, A. (2001). Peer-to-peer: harnessing the power of disruptive technologies. O'Reilly & Associates.

5

Suda, T., Itao, T., Matsuo, M. (2004). The bio-networking architecture: the biologically inspired approach to the design of scalable, adaptive, and survivable/

available network applications. in K. Park (Ed.), *The Internet as a Large-Scale Complex System*, Oxford University Press.

6

Wang, M., and Suda, T. (2001).The bio-networking architecture: a biologically inspired approach to the design of scalable, adaptive, and survivable/available network applications. In *Proceedings of the 1st IEEE Symposium on Applications and the Internet (SAINT)*.

7

Yao, X. (1999). Evolving artificial neural networks. In *Proceedings of the IEEE*, vol. 87.

8

The Bio-Networking Architecture, <**http://netresearch.ics.uci.edu/bionet/**>.

Biographies

Anh Nguyen (**anguyen714@yahoo.com**) received his bachelor degree from the School of Information and Computer Science at University of California, Irvine, 2004. He has worked for the Bio-Networking Architecture project from 2002 to 2004 as an undergraduate researcher. He is currently preparing for graduate study.

Tadashi Nakano (**tnakano@ics.uci.edu**) is a postdoctoral researcher of the School of Information and Computer Science at University of California, Irvine. He received his M. E. and Ph.D. from Osaka University, Japan. His research interests include complex adaptive systems and evolutionary computation.

Tatsuya Suda (**suda@ics.uci.edu**) is a professor of the School of Information and Computer Science at University of California, Irvine. His current research focuses on application of biological principles and large complex system principles onto networks, high speed networks, next generation Internet, ATM (Asynchronous Transfer Mode) networks, object-oriented distributed systems, and multimedia applications.