iNet-EGT: An Evolutionarily Stable Adaptation Framework for Network Applications

Chonho Lee[1], Junichi Suzuki[1], and Athanasios V. Vasilakos[2]

[1] Department of Computer Science
University of Massachusetts, Boston, USA
`{chonho,jxs}@cs.umb.edu`
[2] Department of Computer and Telecommunication Engineering
University of Western Macedonia, Greece
`vasilako@ath.forthnet.gr`

Abstract. This paper studies a bio-inspired framework, iNet-EGT, to build autonomous adaptive network applications. In iNet-EGT, each application is designed as a set of agents, each of which provides a functional service and possesses biological behaviors such as migration, replication and death. iNet-EGT implements an adaptive behavior selection mechanism for agents. It is designed after an immune process that produces specific antibodies to antigens (e.g., viruses) for eliminating them. iNet-EGT models a set of network conditions (e.g., workload and resource availability) as an antigen and an agent behavior as an antibody. iNet-EGT allows each agent to autonomously sense its surrounding network conditions (an antigen) and select a behavior (an antibody) according to the conditions. This behavior selection process is modeled as a series of evolutionary games among behaviors. It is theoretically proved to converge to an evolutionarily stable (ES) equilibrium; a specific (i.e., ES) behavior is always selected as the most rational behavior against a particular set of network conditions. This means that iNet-EGT allows every agent to always perform behaviors in a rational and adaptive manner. Simulation results verify this; agents invoke rational (i.e., ES) behaviors and adapt their performance to dynamic network conditions.

Key words: Artificial immune systems, Evolutionary game theory, Biologically-inspired networking, Autonomous and adaptive networks

1 Introduction

Network applications face critical challenges such as *autonomy*–the ability to operate with minimal human intervention and *adaptability*–the ability to adjust their operations to dynamic changes in network conditions such as workload and resource availability. In order to address these challenges, this paper investigates a biologically-inspired framework to design autonomous adaptive network applications. Based on an observation that various biological systems (e.g., bee colonies) have successfully attained autonomy and adaptability, the authors of

the paper believe that, if network applications are designed after key biological mechanisms, they may be able to attain autonomy and adaptability as well.

In this paper, each network application is designed as a decentralized group of software agents. This is analogous to a bee colony (an application) consisting of multiple bees (agents). Each agent implements a functional service and follows biological behaviors such as migration, replication and death. This paper focuses on an adaptive behavior selection mechanism for agents. The proposed mechanism, called iNet-EGT, is designed after immunological *antigen-antibody reaction*, which produces antibodies specific to antigens (e.g., viruses) for eliminating them. iNet-EGT models a set of network conditions (e.g., workload and resource availability) as an antigen and an agent behavior as an antibody. Each agent contains iNet-EGT as its behavior selection mechanism. iNet-EGT allows each agent to autonomously sense its surrounding network conditions (an antigen) and select a behavior (an antibody) suitable for the sensed conditions. For example, agents may invoke the replication behavior at the network hosts that accept a large number of user requests for their services. This leads to the adaptation of agent availability; agents can improve their throughput.

In iNet-EGT, antigen-antibody reaction (i.e., behavior selection) process is modeled with evolutionary game theory. Each agent contains a set (or population) of behaviors. In a behavior selection process, randomly-selected two behaviors play a game. Each game distinguishes a winning and a losing behavior according to their payoff values computed based on the current network conditions. The winner replicates itself and increases its share in the population. The loser disappears in the population. Through multiple games performed repeatedly in the population the population state (behavior distribution) changes. Through theoretical analysis, iNet-EGT guarantees that the population state converges to an equilibrium where the population is occupied by only one type of behaviors, called strictly dominant behaviors. Each agent invokes a strictly dominant behavior as the most rational behavior against the current network conditions.

iNet-EGT theoretically proves that the population state is evolutionarily stable (ES) when it is on an equilibrium. An ES state is the state that, regardless of the initial population state, the population state always converges to. In this state, no other behaviors except strictly dominant one can dominate the population. Given this property, iNet-EGT guarantees that all agents deterministically invoke a specific ES behavior under a particular set of network conditions. Simulation results verify this theoretical analysis; agents seek equilibria to invoke ES behaviors and adapt their performance to dynamic network conditions.

2 Backgroud: Evolutionary Game Theory

Game theory studies strategic selection of behaviors in interactions among rational players. In a game, given a set of strategies, each player strives to find a strategy that optimizes its own payoff depending on the others' strategy choices. Game theory seeks such strategies for all rational players as a solution, called

Nash equilibrium (NE), where no players can gain extra payoff by unilaterally changing his strategy.

Evolutionary game theory (EGT) is an application of game theory to biological contexts to analyze population dynamics and stability in biological systems. In EGT, games are played repeatedly by players randomly drawn from the population [1, 2]. In general, EGT considers two major evolutionary mechanisms: *mutation*, which injects varieties on genes, and *selection*, which favors some varieties over others based on their fitness to the environment. Mutation is considered in the notion of evolutionarily stable strategies (ESS), which is a refinement of NE. Selection is considered in the replicator dynamics (RD) model.

2.1 Evolutionarily Stable Strategies

ESS is a key concept in EGT. A population following such a strategy is invincible. Specifically, suppose that the initial population is programmed to play a certain pure or mixed strategy x (the incumbent strategy). Then, let a small population share of players $\epsilon \in (0, 1)$ play a different pure or mixed strategy y (the mutant strategy). Hence, if a player is drawn to play the game, the probabilities that its opponent plays the incumbent strategy x and the mutant strategy y are $1 - \epsilon$ and ϵ, respectively. The player's payoff of such a game is the same as that of a game where the player plays the mixed strategy $w = \epsilon y + (1 - \epsilon)x$. The payoffs of players with strategies x and y given that the opponent adopts strategy w are denoted by $U(x, w)$ and $U(y, w)$, respectively.

Definition 1. *A strategy x is called evolutionarily stable if, for every strategy $y \neq x$, a certain $\bar{\epsilon} \in (0, 1)$ exists, such that the inequality*

$$U(x, \epsilon y + (1 - \epsilon)x) > U(y, \epsilon y + (1 - \epsilon)x) \tag{1}$$

holds for all $\epsilon \in (0, \bar{\epsilon})$.

In the special case where the payoff function is linear, $U(x, w)$ and $U(y, w)$ can be written as the expected payoffs for players with strategies x and y, and Equation (1) yields

$$(1 - \epsilon)U(x, x) + \epsilon U(x, y) > (1 - \epsilon)U(y, x) + \epsilon U(y, y) \tag{2}$$

If ϵ is close to zero, Equation (2) yields either

$$U(x, x) > U(y, x), \quad \text{or} \tag{3}$$

$$U(x, x) = U(y, x) \text{ and } U(x, y) > U(y, y) \tag{4}$$

Hence, it becomes obvious that an ESS must be a NE; otherwise, Equation (3) or (4) do not hold.

2.2 Replicator Dynamics

The replicator dynamics, first proposed by Taylor and Jonker [4], specifies how population shares associated with different pure strategies evolve over time. In replicator dynamics players are programmed to play only pure strategies. To define the replicator dynamics, consider a large but finite population of players programmed to play pure strategy $k \in K$, where K is the set of strategies. At any instant t, let $\lambda_k(t) \geq 0$ be the number of players programmed to play pure strategy k. The total population of players is given by $\lambda(t) = \sum_{k \in K} \lambda_k(t)$. Let $x_k(t) = \lambda_k(t)/\lambda(t)$ be the fraction of players using pure strategy k at time t. The associated population state is defined by the vector $\mathbf{x}(t) = [x_1(t), \cdots, x_k(t), \cdots, x_K(t)]$. Then, the expected payoff of using pure strategy k given that the population is in state $\mathbf{x}$ is $U(k, \mathbf{x})$ and the *population average payoff*, that is the payoff of a player drawn randomly from the population, is $U(\mathbf{x}, \mathbf{x}) = \sum_{k=1}^{K} x_k \cdot U(k, \mathbf{x})$. Suppose that payoffs are proportional to the reproduction rate of each player and, furthermore, that a strategy profile is inherited. This leads to the following dynamics for the population shares x_k

$$\dot{x}_k = x_k \cdot [U(k, \mathbf{x}) - U(\mathbf{x}, \mathbf{x})] \tag{5}$$

where x_k is the time derivative of x_k. The equation states that populations with better (worse) strategies than average grow (shrink). However, there are cases when even *a strictly dominated strategy* may gain more than average. Hence, it is not a priori clear whether if such strategies get wiped out in the replicator dynamics. The following theorem answers this question [1]:

Theorem 1. *If a pure strategy k is strictly dominated then $\xi_k(t, x^0)_{t \to \infty} \to 0$, where $\xi_k(t, x^0)$ is the population at time t and x^0 is the initial state.*

On the other hand, it should be noted that the ratio x_k/x_ℓ of two population shares $x_k > 0$ and $x_\ell > 0$ increases with time if the strictly dominated strategy k gains a higher payoff than the strictly dominated strategy ℓ. This is a direct result of Equation (5) and may be expressed analytically via

$$\frac{d}{dt} \left[\frac{x_k}{x_\ell} \right] = [U(k, \mathbf{x}) - U(\ell, \mathbf{x})] \frac{x_k}{x_\ell} \tag{6}$$

¿From Equation (6), it is evident that even suboptimal strategies could temporarily increase their share before being wiped out in the long run. However, there is a close connection between NE and the steady states of the replicator dynamics, which is states where the population shares do not change their strategies over time. Thus, since in NE all strategies have the same average payoff, every NE is a steady state. The reverse is not always true: Steady states are not necessarily NE, e.g., any state where all players use the same pure strategy is a steady state, but, it is not stable [1].

In this paper, a single fixed-sized population model is used; also, discrete time (i.e., generational) model is assumed.

3 iNet-EGT

This section describes how iNet-EGT is designed after an immunological process.

3.1 The Natural Immune System

The immune system is an adaptive defense mechanism to regulate the body against dynamic environmental changes such as antigen invasions. Through a number of interactions among various white blood cells (e.g., macrophages and lymphocytes) and molecules (e.g., antibodies), the immune system evokes *antigen-antibody reaction* to produce antibodies specific to detected antigens. In each interaction, an antibody stimulates or suppresses another one based on its affinity to an antigen.

Through the stimulation and suppression relationships, antibodies dynamically change their population. For example, a stimulated/suppressed antibody replicates/dies and increases/decreases its population. The population of specific antibodies rapidly increases following the recognition of an antigen and decreases again after eliminating the antigen. Through this self-regulation mechanism, adaptive immune response is an emergent product of interactions among antibodies.

3.2 Immunologically-inspired Adaptation Behavior Selection

An agent contains iNet-EGT as its own immune system. iNet-EGT implements an adaptive behavior selection mechanism for an agent by following antigen-antibody reaction in the natural immune system. It is designed to allow each agent to autonomously sense a set of its surrounding network conditions (an antigen) and adaptively perform a behavior (an antibody) suitable for the conditions (Figure 1).

In iNet-EGT, an antigen consists of network conditions: $C = \{c_1, c_2, \cdots, c_L\}$ where L denotes the number of network conditions that each agent senses. For example, $C = \{100 : Workload, 35 : ResourceUtilization\}$ may mean 100 user requests per minute as workload and 35% memory utilization as resource utilization. Each antibody represents one of behavior types (e.g., migration, replication and death): $B = \{b_1, b_2, \cdots, b_M\}$ where M denotes the number of behavior types that each agent invokes.

In iNet-EGT, behavior selection (i.e., antigen-antibody reaction) is modeled based on evolutionary game theory. iNet-EGT executes an evolutionary game in a population of behaviors (antibodies) and determines one of the behaviors to be invoked by an agent. After initializing the population, randomly-selected two behaviors repeatedly play games in the population. Each game distinguishes a winning behavior and a losing behavior according to their fitness (or payoff) values that are computed based on the current network conditions. The loser disappears in the population. The winner replicates itself and increases its share in the population. The winner is also mutated at a certain probability in order to
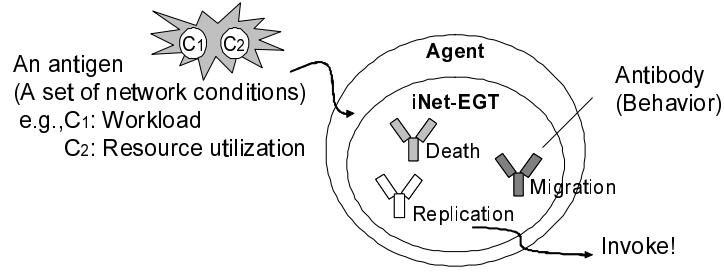
Fig. 1. Antigen-antibody reaction

react to future changes in network conditions. Then, eventually one behavior in a population is selected as the behavior invoked by an agent. Figure 2 presents a pseudocode of the implemented behavior selection process.

BehaviorSelection()
// P: Population, W: A set of winners, M: A set of the mutated
// a_b: A behavior invoked by an agent
main
  INITIALIZEPOPULATION(P)
  while (the termination condition is not satisfied)
  $$\mathbf{do} \begin{cases} W, M \leftarrow \Phi, \; n \leftarrow P/2 \\ \mathbf{for} \; i \leftarrow 0 \; \mathbf{to} \; n \\ \quad \mathbf{do} \begin{cases} \text{COMPUTEFITNESSVALUE}(P) \\ \{behavior_1, behavior_2\} \leftarrow \text{SELECT}(P) \\ P \leftarrow P - \{behavior_1, behavior_2\} \\ winner \leftarrow \text{PERFORMGAME}(behavior_1, behavior_2) \\ W \leftarrow W \cup winner \\ M \leftarrow M \cup \text{MUTATE}(winner) \end{cases} \\ P \leftarrow W \cup M \end{cases}$$
  $a_b \leftarrow \{ \text{b} \mid \text{b} \in P, \; X_b(t) > th \}$

Fig. 2. Pseudocode of behavior selection in iNet-EGT

Figure 3 describes how behavior selection process works in each generation. A population is implemented as an array of behaviors, each behavior is associated with one of behavior types (i.e., actual actions as strategies) such as migration and replication. A population state at time t represents behavior distribution in the population, and it is denoted by $X(t) = \{x_1(t), x_2(t), \cdots, x_M(t)\}$ where $x_b(t)$ is the population share of a behavior type b, i.e., $x_b = \frac{n_b}{N}$, where $N = \sum_{b \in B} n_b$ where n_b is the number of behaviors with a behavior type b; so $\sum_{b \in B} x_b = 1$. Initially, behavior types are evenly distributed into behaviors in a population. For example, if the size of population is 100 and the size of behavior set is 4, then each 25 behaviors has one of the behavior types, i.e., $X(0) = \{.25, .25, .25, .25\}$.

Each behavior (antibody) in a population computes its own fitness value (as affinity) against a set of network conditions (antigen). F_b denotes a fitness value
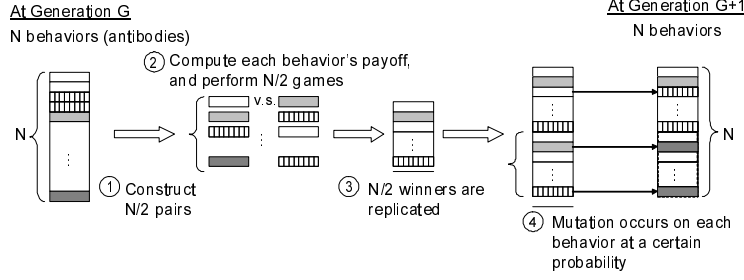
Fig. 3. Behavior Selection in each Generation

of behavior b. In the current study, 4 behaviors (M: migration, R: replication, D: death, N: do-nothing) and 2 network conditions $C = \{c_1, c_2\}$ are considered. F is designed as follows:

$$F_{M^k} = c_1^k + c_2^k, \quad F_D = (1 - c_1) - c_2$$
$$F_R = c_1 + c_2, \quad F_N = (1 - c_1) + (1 - c_2)$$

Two network conditions, *Queue length* (c_1) and *Request rate* (c_2), indicate the spatial and temporal changes of the network environment in terms of workload. Queue length c_1^k is the number of user requests waiting to be processed in a queue at node k. Request rate c_2^k is the difference between the number of user requests received for a particular time period and that for the previous time period. Assume that a node maintains the number of user requests, $R(\Delta T)$, for a time period between $t - 1$ and t. Request rate is computed as $c_2(t) = R(\Delta(T)) - R(\Delta(T - 1))$.

A game is performed between randomly paired behaviors. A behavior wins/loses against another one based on their fitness values. A losing behavior is removed from a population. A winning behavior survives for the next generation and makes its copy to increase its population share; in addition, the mutation occurs on each copied behavior at a certain probability to change its behavior to another. iNet-EGT repeats the same process until the termination condition is satisfied. When one of the behaviors occupies the population based on the condition, $X_b(t) > th$, the behavior type b is selected. The threshold value th is set to 0.95 since the mutation probability is set to 0.05 in simulation studies.

4 Stability Analysis

This section analyzes the stability of behavior selection in iNet-EGT by showing that a population state converges to an evolutionarily stable state (or an asymptotically stable state) in three steps: (1) The dynamics of population state change over time is formalized as a set of differential equations, (2) The proposed behavior selection has equilibrium points, (3) The equilibrium points are asymptotically stable. First, in order to construct the differential equations, following terminologies and variables are defined.

– B denotes a set of behavior types. $B = \{b_1, b_2, \cdots, b_M\}$, and M denotes the number of behavior types.
– N denotes a population size. $N = \sum_{b \in B} n_b$ where n_b is the number of behaviors with a behavior type b.
– $X(t)$ denotes a population state at time t. $X(t) = \{x_1(t), x_2(t), \cdots, x_M(t)\}$ where x_b is the population share of a behavior type b ($x_b = \frac{n_b}{N}$; $\sum_{b \in B} x_b = 1$).
– F_b is the fitness value of a behavior with a behavior type b.
– p_k^b denotes the probability that a behavior with a behavior type b is replicated by winning a game against the behavior with a behavior type k. It is computed by $p_k^b = x_b \cdot \phi(F_b - F_k)$ where $\phi(F_b - F_k)$ is the conditional probability that the fitness value of a behavior with a behavior type b is larger than that of a behavior type k.

How behaviors with a behavior type b change their population share is considered as the sum of difference between the number of behaviors which are replicated (win) and eliminated (lose) at a time; then it is formalized as follows (using a brevity $c_{bk} = \phi(F_b - F_k) - \phi(F_k - F_b)$).

$$\dot{x}_b = \sum_{k \in B, k \neq b} \{x_k p_k^b - x_b p_b^k\} = x_b \sum_{k \in B, k \neq b} x_k \{\phi(F_b - F_k) - \phi(F_k - F_b)\}$$

$$= x_b \sum_{k \in B, k \neq b} x_k \cdot c_{bk} \tag{7}$$

Theorem 2. *If a behavior with a behavior type k is strictly dominated, then $x_k(t) \to 0$ as $t \to \infty$.*

In game theory, it is said that a strategy (behavior type) is strictly dominant if, regardless of what any other players (behaviors) select, a player with the strategy gains a strictly higher payoff than any others. If a behavior has a strictly dominant behavior type, than it is always better than any others in terms of a fitness value (payoff). It will increase its population share and occupy a population over time. So, if a behavior is strictly dominated, then the behavior disappear in a population over time.

Theorem 3. *The population state of an agent converges to an equilibrium.*

Proof. It is true that, according to the fitness function (Equation 7), behaviors with different behavior types have different fitness values under the same network conditions. In other words, under the particular network conditions, only one behavior has the highest fitness value among the others. Assume that $F_1 > F_2 > \cdots > F_M$, and by Theorem 1, a population state eventually converges to $X(t) = \{x_1(t), x_2(t), \cdots, x_M(t)\} = \{1, 0, \cdots, 0\}$ as an equilibrium. Differential equations should satisfy the constraint $\sum_{b \in B} x_b = 1$.   □

Theorem 4. *The equilibrium of behavior selection in iNet-EGT is evolutionarily stable (i.e., asymptotically stable).*

Proof. At the equilibrium where $X = \{1, 0, \cdots, 0\}$, a set of differential equations can be rewritten in the downsized by substituting $x_1 = 1 - x_2 - \cdots - x_M$

$$\dot{z}_b = z_b[c_{b1}(1 - z_b) + \sum_{i=2, i \neq b}^{M} z_i \cdot c_{bi}] \quad \text{where} \quad \text{b = 2,...,M} \tag{8}$$

where $Z(t) = \{z_2(t), z_3(t), \cdots, z_M(t)\}$ denotes the corresponding downsized population state, which is an equilibrium $Z_{eq} = \{0, 0, \cdots, 0\}$ of (M-1)-dimension based on Theorem 2.

To verify that a state at the equilibrium is an asymptotically stable state, show that all the Eigenvalues of Jaccobian matrix of the downsized population state has negative Real parts. The elements of Jaccobian matrix J are

$$J_{bk} = \left[\frac{\partial \dot{z}_b}{\partial z_k}\right]_{|Z=Z_{eq}} = \left[\frac{\partial z_b[c_{b1}(1 - z_b) + \sum_{i=2, i \neq b}^{M} z_i \cdot c_{bi}]}{\partial z_k}\right]_{|Z=Z_{eq}} \tag{9}$$

$$where\ b, k = 2, ..., M$$

Therefore, Jaccobian matrix J is given by

$$J = \begin{bmatrix} c_{21} & 0 & \cdots & 0 \\ 0 & c_{31} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & c_{M1} \end{bmatrix} \tag{10}$$

where $c_{21}, c_{31}, \cdots, c_{M1}$ are the Eigenvalues of J. According to Theorem 2, $c_{b1} = -\phi(F_1 - F_b) < 0$ for every b; therefore, $Z_{eq} = \{0, 0, \cdots, 0\}$ is asymptotically stable. An agent deterministically invokes a specific behavior (i.e., ES behavior) under a particular set of network conditions. $\square$

5 Simulation Results

This section evaluates iNet-EGT through simulations. Figure 4 shows a simulated network, which is a server farm consisting of 16 (4 x 4) hosts in a grid topology. User requests travel from users to agents via user access point. This simulation study assumes that a single (emulated) user runs on the access point and sends user requests to agents.

At the beginning of a simulation, an agent is deployed on a randomly-selected host in the network. Each agent has its own iNet-EGT that contains a population
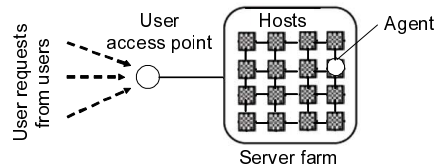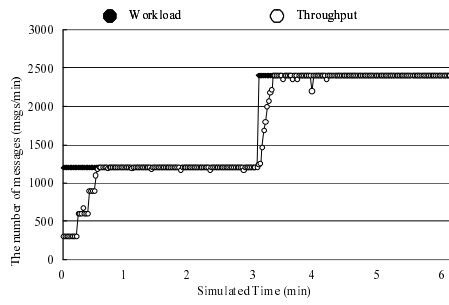

Fig. 4. Simulated Network

Fig. 5. Workload Type 1 and Throughput
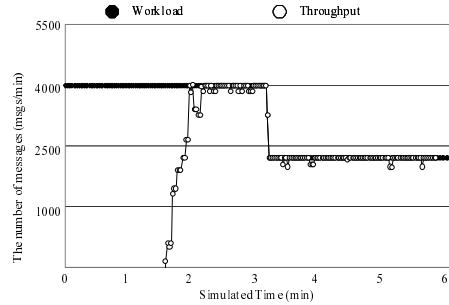

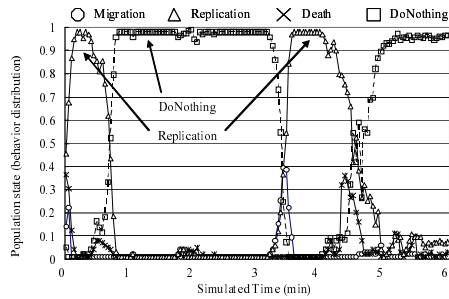
Fig. 6. Workload Type 2 and Throughput

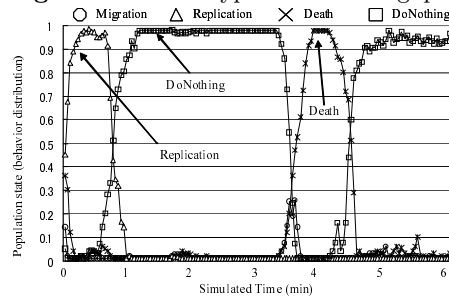**Fig. 7.** Population State (Workload #1)

Fig. 8. Population State (Workload #2)

of 100 behaviors. (25 behaviors are of each of four behavior types: migration, replication, death and do-nothing). Mutation rate and behavior selection threshold are set to 0.1 and 0.95, respectively. Figures 5 and 6 show two different types of changes in workload (i.e., the number of user requests) given to agents.

Figure 7 shows how population state (behavior distribution) changes over time in an agent deployed at the beginning of a simulation. (The two figures show the changes in population state against the workload type 1 and 2, respectively.) In Figure 7, the number of replication behaviors increases in the first 15 seconds, and population state converges to an ES state. Then, the do-nothing behavior takes over the replication behavior to dominate the population; the population converges to another ES state. The second ES state emerges because agents finish adapting their availability with the replication behavior in the first one minute to efficiently process incoming user requests. (See Figure 9 for the changes in agent availability under the workload type 1.) This ES state continues until workload spikes at the third minute. Upon the workload spike, the replication behavior dominates the behavior population again. Once agent availability adapts to the workload spike, the do-nothing behavior takes over the replication behavior.

Figure 8 shows the changes in population state under the workload type 2. The changes are similar to those in Figure 7 except that the death behavior dominates the behavior population when workload drops. See Figure 10 for the changes in agent availability under the workload type 2. As shown in Figures 7 and 8 , iNet-EGT allows agents to successfully seek ES equilibria in their behavior selection according to dynamic network conditions.
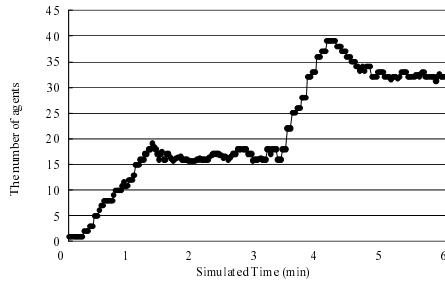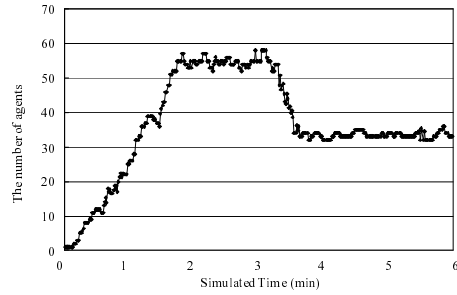
Fig. 9. Agent Availability (Workload #1)

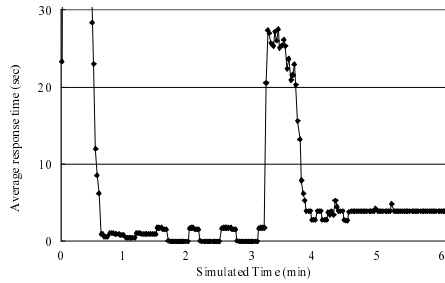
Fig. 10. Agent Availability (Workload #2)

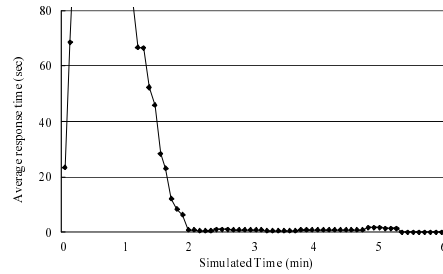**Fig. 11.** Response Time (Workload #1)

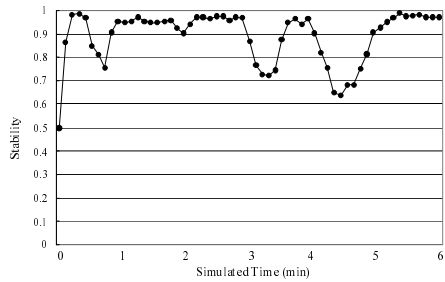**Fig. 12.** Response Time (Workload #2)
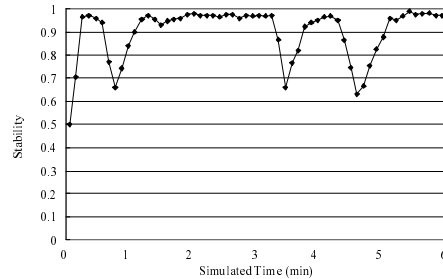

Fig. 13. Stability (Workload #1)

Fig. 14. Stability (Workload #2)

Figures 9 and 10 show how agent availability (i.e., the number of agents) changes over time when the workload type 1 and 2 are given to agents, respectively. The two figures demonstrate that agents adapt their availability by invoking behaviors according to the ES states they are on. (See also Figures 7 and 8.)

Figures 5 and 11 show the throughput (i.e., the number of processed requests per minute) and response time that agents yield for users under the workload type 1. Figures 6 and 12 show the throughput and response time results under the workload type 2. At the beginning of a simulation, only one agent is deployed; it cannot efficiently process all user requests. As a result, throughput is low, and response time is high. However, as agents performs their behaviors by seeking

ES states (Figures 7 and 8), they adapt their throughput and response time to dynamic network conditions.

Figure 13 and 14 show the average stability of the behavior populations that agents possess. It is measured as follows:

$$S_{avg}(t) = \frac{1}{A(t)} \sum_i \max_{b \in B}\{x_b(t)\} \tag{11}$$

where $A(t)$ denotes the total number of agents. i indexes agents. b indexes behavior types $(1,\cdots,4)$. Agents seek equilibria to invoke evolutionarily stable behaviors. For example, when agents sufficiently adapt their availability to the workload at around 0:30, $S_{avg}(t)$ decreases because the number of replication and do-nothing behaviors change. However, soon or later agents increase the number of do-nothing behaviors, and a population state converges to the stable state again. In addition, the likelihood of agents operating at a stable state during a simulation run (e.g., 6 min) is observed as *stability*. It is measured as how long agents operate at equilibria during the simulation run (i.e., [time(sec) for $S_{avg}(t) > 0.95$]/[6 min=360 sec]). Along the workload type 1, the stability is about 82%. For the workload type 2, the stability is about 86%.

6 Related Work

iNet-EGT is an extension to its predecessor called iNet [6]. iNet-EGT and iNet share the same goal; immunologically-inspired adaptive behavior selection for agents. However, they are different in their approaches to design antigen-antibody reaction and antibody evolution. iNet designs antigen-antibody reaction based on a model built with the immune network hypothesis [5] and designs antibody evolution with a genetic algorithm. iNet-EGT takes evolutionary game theoretic approach to design antigen-antibody reaction and antibody evolution. It guarantees stability in behavior selection while iNet does not. iNet-EGT is the first attempt to model an artificial immune system based on EGT.

Conventional game theory has been introduced to several aspects in network systems; e.g., job allocation [7], security [8–10] and routing [11]. They focus on the rationality of behavior selection in static network environments; however, they do not consider adaptation in dynamic network environments. [12,13] leverage EGT to formulate rational and adaptive routing decisions to dynamic network environments. Unlike [12,13], iNet-EGT performs the mutation operation in the behavior selection to better adapt to future changes in network environments.

[14–17] study adaptive behavior selection mechanisms for agent-based systems. [14] proposes a rule-based mechanism, which is similar to iNet-EGT in that it implements deterministic behavior selection. However, unlike [14], iNet-EGT guarantees stability in behavior selection. [15–17] consider non-deterministic behavior selection with stochastic algorithms. In contrast, iNet-EGT considers determinism in behavior selection to guarantee its stability.

7 Conclusion

This paper proposes and evaluates a bio-inspired framework, iNet-EGT, which aids building autonomous and adaptive network applications. iNet-EGT is designed after antigen-antibody reaction in the immune system. The reaction process is modeled as a series of evolutionary games among behaviors. It is theoretically proved to converge to an evolutionarily stable (ES) equilibrium. This means that iNet-EGT allows every agent to always perform behaviors in a rational and adaptive manner. Simulation results verify this; agents invoke rational (i.e., ES) behaviors and adapt their performance to dynamic network conditions.

References

1. J. W. Weibull. *Evolutionary Game Theory*. MIT Press, 1996.
2. M. A. Nowak. *Evolutionary Dynamics: Exploring the Equations of Life*. Harvard University Press, 2006.
3. D. Fudenberg and D.K. Levin. *The theory of learning in games*. MIT Press, 1998.
4. P. Taylor and L. Jonker. *Mathematical Biosciences*, 16.
5. N. K. Jerne. Idiotypic networks and other preconceived ideas. *Immunological Review*, 1984.
6. C. Lee, H. Wada, and J. Suzuki. Towards a biologically-inspired architecture for self-regulatory and evolvable network applications. In F. Dressler and I. Carreras, editors, *Advances in Biologically Inspired Information Systems*. Springer, 2007.
7. R. Subrata, A. Y. Zomaya, and B. Landfeldt. Game theoretic approach for load balancing in computational grids. *IEEE Transactions on Parallel and Distributed Systems*, 19(1), 2008.
8. M. Kodialam and T.V.Lakshman. Detecting network intrusions via sampling: a game theoretic approach. In *Proc. of IEEE INFOCOM*, April 2003.
9. A. Agah, K. Basu, and S.K. Das. Preventing dos attack in sensor networks: a game theoretic approach. In *Proc. of IEEE ICC*, May 2005.
10. H. Otrok, M. Mehrandish, and et al. C. Assi. Game theoretic models for detecting network intrusions. *Computer Communications*, 31, June 2008.
11. R. Kannan and S. Iyengar. Game theoretic models for reliable path-length and energy constrained routing with data aggregation in wireless sensor networks. *IEEE J. on Selected Areas in Communications*, 22(6), 2004.
12. A.V.Vasilakos and M. Anastasopoulos. Application of evolutionary game theory to wireless mesh networks. In *Studies in Comp. Intelligence*. Springer, 2007.
13. M. P. Anastasopoulos, D. K. Petraki, R. Kannan, and A. V. Vasilakos. Tcp throughput adaptation in wimax networks using replicator dynamics. *IEEE Transactions on Systems, Man, and Cybernetics*, to appear.
14. Z. Li and M. Parashar. Rudder: A rule-based multi-agent infrastructure for supporting autonomic grid applications. In *Proc. of IEEE ICAC*, 2004.
15. Y. Wang, S. Li, Q. Chen, and Weili Hu. Biology inspired robot behavior selection mechanism: Using genetic algorithm. In *Proc. of LSMS*, 2007.
16. B. D. Damas and L. Custódio. Emotion-based decision and learning using associative memory and statistical estimation. *Informatica*, 27(2), 2003.
17. K-J. Kim and S-B. Cho. Bn+bn: Behavior network with bayesian network for intelligent agent. In *Proc. of Australian Conf. on Artificial Intelligence*, 2003.