A trade-off between agility and resilience*

Erol Gelenbe and Yu Wang

Dept. of Electrical and Electronic Engineering
Imperial College
London, SW7 2BT, UK
{e.gelenbe,yu.wang3}@imperial.ac.uk

Abstract. In many natural or artificial settings, agility and resilience are contradictory properties of agents and populations. The settings one can imagine for this contradiction include very different objects such as computer software (agile code is small, runs fast but is more vulnerable to attacks and failures, while resilient code can be bulky and run slowly but has built-in redundancy and robustness to all kinds of attacks and system failures), military units (non-armored versus heavily armored units), commercial companies (small and lean, or large and slow), and many other situations of interest. In this paper we discuss a mathematical model which represents a population of agents in the presence of other malicious agents, and addresses the trade-off between (a) providing resilience to objects which gives them greater ability to resist attacks but reduces their mobility to attack and destroy their potential attackers, and (b) leaving them in a more agile state where they can more effectively search and destroy their potential attackers, but where they are themselves more vulnerable to attack.

1 Introduction

Agility and resilience are often contradictory properties of agent populations. For instance, in computer software small and light-weight agile code may require a small amount of memory and it will run fast, but it will not contain all the additional elements which make it less vulnerable to attacks and failures. On the other hand, resilient code may be large and bulky and run slowly, but will have built in redundancy and robustness to all kinds of attacks and system failures, such as anti-viral software, audit trails and roll-back-recovery mechanisms.

In a different setting, light cavalry units may be very effective to rapidly engage and destroy the enemy but will be more vulnerable to attacks, while heavy armored units may be well protected in a relatively static situation, but could be very ineffective in difficult terrains where they would become easy targets for well armed and mobile adversaries. In the business world, large commercial companies may have vast cash reserves which can allow them to resist attacks on their existing markets by lowering

* This work was supported by a contract from General Dynamics UK Ltd. to Imperial College under DIF DTC Project 6.8.

prices, but they may be poorly equipped to invent new products and rapidly enter new markets where much smaller and flexible start-ups can be more effective.

In network centric software environments, it is worth considering the trade-off between having small agile agents which can search and destroy sources of viral or worm attacks, as well as carry out their primary computational functionality, and large and bulky agents which are heavily protected but cannot travel through the system rapidly to avoid, or search and destroy potential attackers.

In this paper we discuss a model of two competing populations, which we call the *viruses* and the *agents*. A virus may indiscriminately attack an agent, and when it does this it will either destroy the agent or be itself destroyed. On the other hand, agents can be of two types: Type A (agile) agents have no immunity against viruses so if attacked they have a higher chance of being destroyed than Type B (heavy) agents, which have been immunized and therefore have a smaller chance of being destroyed by a virus attack. Type B agents are just Type A agents which have been immunized. On the other hand, if a Type A agent attacks the virus population, it has a high probability of destroying a virus and of escaping unscathed, while when a Type B agent attacks the viruses it has a smaller probability of successfully destroying a virus and a larger probability of being destroyed in the process.

In the following section we present a mathematical model of this system in terms of infinite discrete state space continuous time Markov processes. We show that the equilibrium probability distribution for the system can be obtained using the theory of G-Networks [2, 3, 4]. This then allows us to consider various parameter optimization issues in Section 3, where we also present several numerical examples.

2 A Model for Two Competing Populations with Immunization

The state of the system under consideration will be the vector $x(t) = (x_V(t),x_A(t),x_B(t))$, where the elements of the vector are the numbers of viruses, Type A agents and Type B agents at time t, respectively. Note that these (discrete) numbers may also refer in certain cases to concentrations of agents (e.g. biological viruses per cubic meter, tens of computer viruses per computer node, and so on).

Agents arrive to the system in a Poisson process of rate λ_A, while viruses arrive in another Poisson process of rate λ_V, both of these processes being independent of each other Note that the external arrivals of agents only concern Type A agents, since a Type B agent is obtained by modifying (i.e. immunizing) a Type A agent. Whenever there are viruses in the system, they will attack the population of agents at rate R_V, and will select a victim among the Type A agents with probability α, or among the Type B agents with probability β ($\alpha + \beta = 1$). Before deciding to attack the agents, the viruses do not verify if there are any agents available; so that the virus will be

destroyed after any attempted attack. Thus, as a result of a virus attack, the attacking virus itself will be destroyed, and when attacked a Type A agent will be "killed" with probability c, while a Type B agent will be killed with probability K.

Concurrently with this, provided Type A agents exist, they take action at rate R_A either by attacking a virus with probability q, or they choose to immunize themselves and thus mutate into a Type B agent with probability (1-q). If a Type A agent attacks a virus, it will successfully destroy it with probability a, in which case either it will return unscathed to rejoin the population of Type A agents with probability e or it will itself be destroyed as well as the virus it attacked. With probability (1-a) when a Type A agent attacks a virus it is unsuccessful and is itself destroyed. A slightly different model (which we do not consider here but will be developed in further work) includes an "immunization server" that a Type A agent must visit in order to mutate into a Type B agent.

Also concurrently, provided Type B agents exist, they take action at rate R_B by attacking a virus. If a Type B agent attacks a virus, it will successfully destroy it with probability b, in which case it either returns unscathed to rejoin the population of Type B agents with probability f; or with probability (1-f) it is itself be destroyed while destroying a virus. With probability (1-b) when a Type B agent attacks a virus, it is unsuccessful and is itself destroyed.

The difference in the capability of Type A and Type B agents with respect to self-protection against virus attacks is given by the probabilities c and K, while their ability to attack successfully is given by the probabilities a and b, and their ability to remain unscathed after an attack is given by the probabilities e and f.

In general we would represent the contrast between agility and resilience of Type A and B agents as follows: $R_A \geq R_B$, e > f, and K > c. The described system can be illustrated by the figure below (See Figure 1).

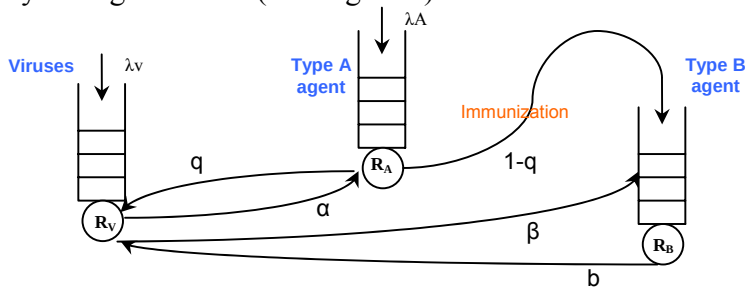

Figure 1 – Two competing populations with immunization

2.1 System Equations

In the sequel we assume that all the rates defined above are parameters of exponential distributions for independent random variables, so that the system we have described

is reduced to an infinite and discrete state space and continuous time Markov chain. We will not detail the derivation of the following equation (1), but simply indicate that the process $\{x(t): t \geq 0\}$ can be studied by writing the Chapman-Kolmogorov equations for $p(k,t) = \text{Prob}[x(t) = k]$, where $k = (k_V, k_A, k_B)$ is a 3-vector of non-negative integers representing the number of viruses, Type A and Type B agents in the system, respectively.

Proposition 1 The Chapman-Kolmogorov equations for the system we have described are:

$$\begin{aligned}
\frac{d}{dt}p(k,t) &= \lambda_V p(k-z_V,t)1[k_V>0] + \lambda_A p(k-z_A,t)1[k_A>0] \\
&+ R_V \alpha c p(k+z_V+z_A,t) + R_V \beta K p(k+z_V+z_B) + R_A q p(k+z_V,t) a e \qquad (1) \\
&+ R_V \alpha p(k+z_V,t)1[k_A=0] + R_V \beta p(k+z_V,t)1[k_B=0] \\
&+ R_V \alpha p(k+z_V,t)(1-c)1[k_A>0] + R_V \beta p(k+z_V)(1-K)1[k_B>0] \\
&+ R_A q p(k+z_V+z_A,t)a(1-e) + R_A q p(k+z_A,t)(1-a) \quad + \quad R_A q p(k+z_A,t)1[k_V=0] \\
&+ R_A(1-q)p(k+z_A-z_B,t)1[k_B>0] + R_B p(k+z_V+z_B,t)b(1-f) \\
&+ R_B p(k+z_V,t)bf + R_B p(k+z_B,t)(1-b) \quad + R_B p(k+z_B,t)1[k_V=0] \\
&- (\lambda_V+\lambda_A)p(k,t) - (R_V 1[k_V>0] + R_A 1[k_A>0] + R_B 1[k_B>0])p(k,t)
\end{aligned}$$

where on the right hand side we show the transition rates from neighboring states into state k, and $z_V = (1,0,0)$, $z_A = (0,1,0)$ and $z_B = (0,0,1)$.

Theorem 2 When the stationary solution of these equations exists, it has the following form [1, 2, 3, 4]:

$$p(k) = \lim_{t \to \infty} p(k,t) = \rho_V^{k_V} \rho_A^{k_A} \rho_B^{k_B}$$

where $k_V \geq 0$, $k_A \geq 0$, $k_B \geq 0$, and the $0 \leq \rho_V, \rho_A, \rho_B < 1$ satisfy the following system of algebraic non-linear equations:

$$\rho_V = \frac{\lambda_V}{R_V + qa\rho_A R_A + b\rho_B R_B}, \qquad \rho_A = \frac{\lambda_A + qae\rho_A R_A \rho_V}{R_A + \alpha c \rho_V R_V}, \qquad \rho_B = \frac{(1-q)\rho_A R_A + bf\rho_B R_B \rho_V}{R_B + \beta K \rho_V R_V}$$

Corollary 3 The total average number of agents in the system in stationary state defined as $N = \lim_{t \to \infty} E[x_A + x_B]$ is given by:

$$N = \frac{\rho_A}{1-\rho_A} + \frac{\rho_B}{1-\rho_B}$$

3 Numerical Examples

In this section we present some of the numerical results obtained from the model under different parameter settings. To simplify matters we take $R_V = R_A = R_B = 1$ and $\lambda_V = \lambda_A = 0.99$ in all of the numerical examples. Through the experiments, we are exploring how different parameters affect the behavior of the model, in this case, the survival of the agents. We therefore use the total average number of agents, N, as the objective function and vary the parameter values.

We firstly examine how the probability of Type A agents attack (q) the viruses effects the average total agent population size. Both α and β are set to be 0.5 to indicate that the viruses attack either population of agents indiscriminately. e and f are set to be 0 to indicate that both of the agents are destroyed after attacking the virus. Varying the strength of the viruses (K), we observed that the total average number of agents decrease as the viruses get stronger as illustrated in the figure (See Figure 2).

In the competing population case, a Type A agent can either be destroyed or escape after a successful attack. We then modify the model to explore how N varies when half of the Type A agents escape after successful attacks (see Figure 3) and e is set to be 0.5 to reflect it. Comparing the two models, N increases dramatically if Type A agents can escape after attacking the viruses and the effect that K has on N remains small.

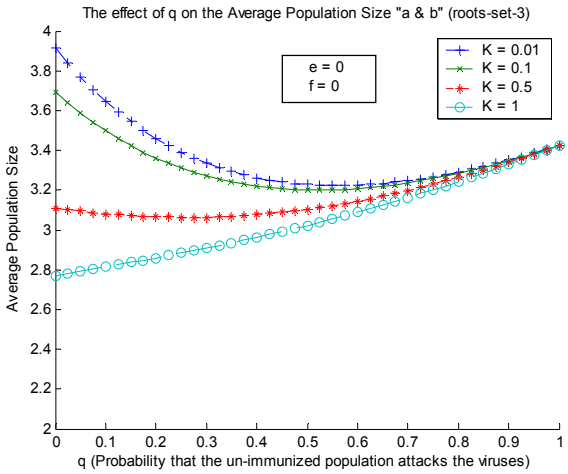

Figure 2 – N varies as a function of the probability that Type A agents attack the viruses before being immunized

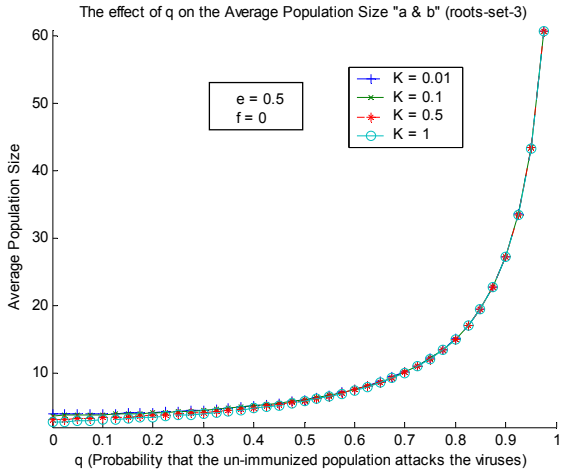

Figure 3 – N as a function of q when Type A agents can escape after attacking the viruses

So far the parameter settings reflect the extreme aspects of the model. Setting e = 1, f = 0.7, a = 0.5, b = 0.3 and plot N against q with the same set of K, we get the figure below (See Figure 4). Differing from the previous models, the current model only has a solution for two sets of K values and within the range of 0 to 0.32 and 0 to 0.72, respectively. As mentioned in the earlier section, the stationary solution of the non-linear equations exists when both $k_V \geq 0$, $k_A \geq 0$, $k_B \geq 0$ and $0 \leq \rho_V$, ρ_A , $\rho_B < 1$ are satisfied. We therefore examine the behavior of ρ_V , ρ_A and ρ_B when K = 1 and c = 0.7. We found that ρ_A approaches 1 as q reaches 0.72 and exceeds 1 after that. When one of the ρ_V , ρ_A or ρ_B reaches 1, the condition is not satisfied and therefore the model does not have a solution after the point q = 0.72. The same applies to ρ_A when K = 0.5 and c = 0.3.

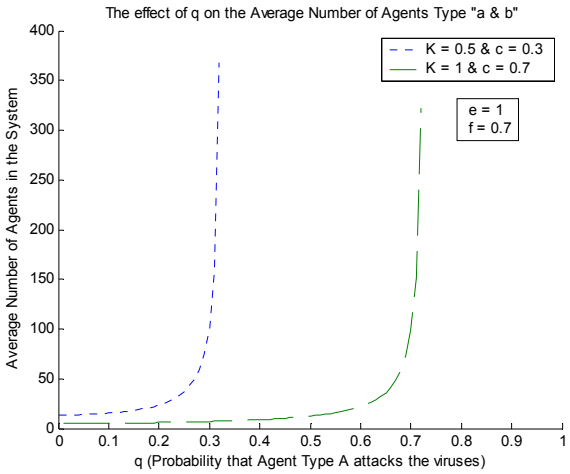

Figure 4 –The behavior of N with different parameter settings

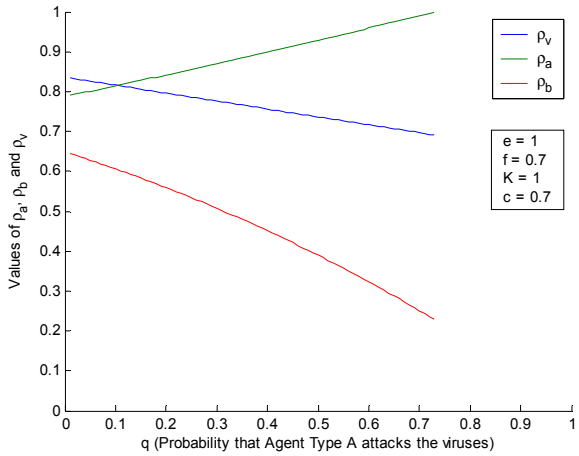

Figure 5 – Behavior of ρ_A, ρ_B and ρ_V

By assigning e = 0.5 and f = 0.3, the model has a solution when K = 0.1 and c = 0.05 as illustrated below (See Figure 6). Comparing Figures 4 and 6, we could draw the conclusion that as e and f increase, the value of N increases drastically and reaches its asymptote at infinity faster. That N increases sharply indicates that ρ_V , ρ_A or ρ_B is reaching 1. By increasing q in much finer steps, we observed that N keeps increasing and approaches infinity; thus the peaks shown in the figures are the maximum values of N obtained by increasing q in steps of 0.01.

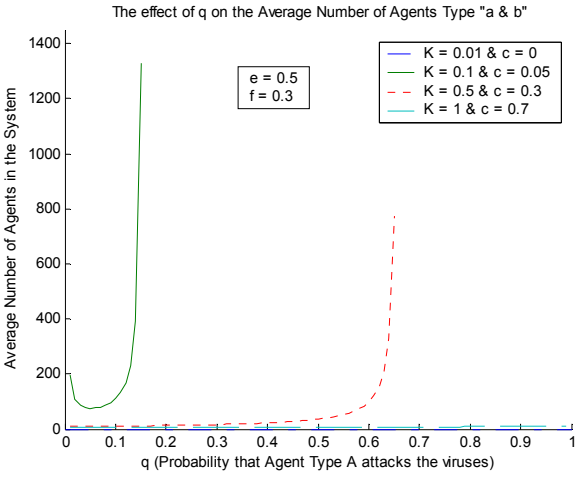

Figure 6 –The behavior of N with different parameter settings

We then modify the model to see the effect of α (the probability that the virus attacks the Type A agents) on N. As illustrated by Figures 7 and 8, N decreases sharply as α increases. This is because the Type A agents are more vulnerable, compared to Type B agents. The effect of e and f on N is not obvious from Figure 7 and 8. In fact, N varied just above 0.3 when the f changes from 0.1 to 0.4.

However, by varying q from 0.3 to 0.7 we see that N changes dramatically as shown in Figure 9. When K = 1 and c = 0.7, the value of N increased from 354 to 58153. The experiments showed that q has a stronger impact on N than α.

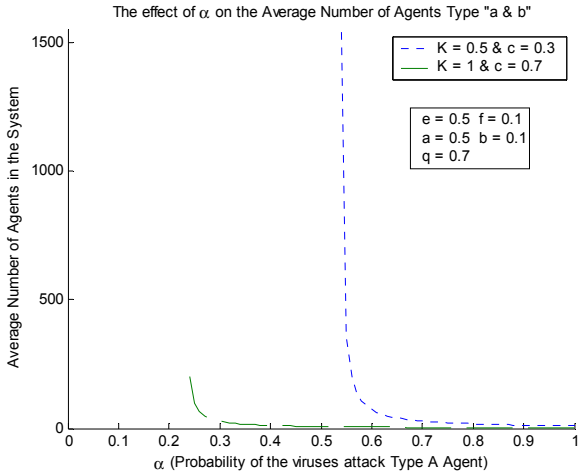

Figure 7 –The behavior of N over a range of α (parameters setting 1)

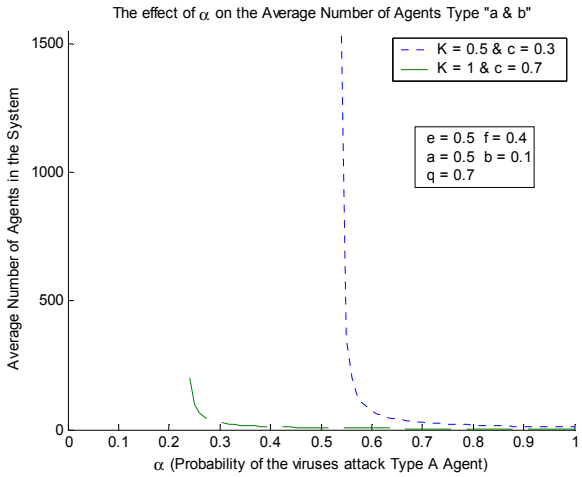

Figure 8 –The behavior of N over a range of α (parameters setting 2)

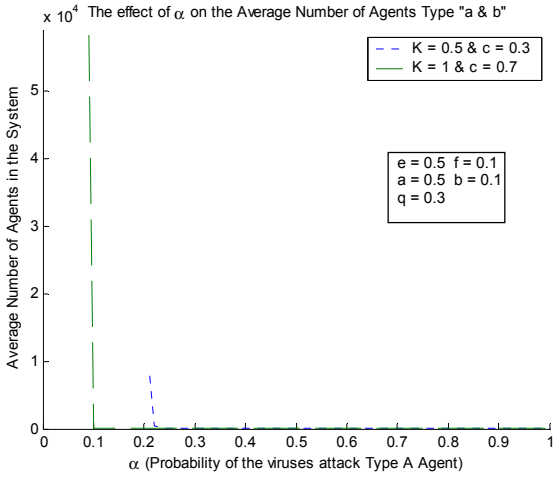

Figure 9 –The behavior of N over a range of α (parameters setting 3)

4 Conclusions and Future Work

This paper has developed a stochastic model of competing agent populations. We have shown how some important characteristics of these populations can impact their ability to survive in a hostile context. Future work will extend this initial approach to more complex models and interactions.

References

1. E. Gelenbe, G-networks with triggered customer movement, Applied Probability, Vol. 30, pp. 742-748. (1993)
2. J.-M. Fourneau, E. Gelenbe and R. Suros, G-networks with multiple classes of negative and positive customers, Theoretical Computer Science, Vol. 155, pp. 141-156. (1996)
3. E. Gelenbe, G-Networks: Multiple Classes of Positive Customers, Signals, and Product Form Results, Performance 2002, pp. 1-16. (2002)
4. E. Gelenbe and J.-M. Fourneau, G-networks with resets, Performance Evaluation, Vol. 49, pp. 179-191. (2002)