Measures of Developing Stability and Perturbations in Ecologies of Semantic Web Services

Elpida S. Tzafestas
Institute of Communication and Computer Systems
National Technical University of Athens
Zographou Campus 15773, Athens, GREECE.
brensham@softlab.ece.ntua.gr

***Abstract*--We are discussing the issue of performance of engineered self-organizing systems. We are arguing that the qualitative concepts of stability, resistance to perturbations, predictability and development, that are drawn from the natural self-organizing systems, can serve the role of measure classes. We subsequently briefly show how these classes can be instantiated to specific measures within an INFRAWEBS application.**

***Index terms*— self-organization, stability, equilibrium, attractor, perturbation, development, virtual organization**

A. Introduction

We are addressing the class of virtual organizations where an entrepreneurial or other activity acts as an interface between a number of existing remote services or applications and a number of customers that interact with them in complex ways. For example, such activities include private e-government middle services that specialize in carrying out complex tasks with many different state administration services, or commercial houses specializing in centralizing and organizing large complicated orders that involve a number of different vendors scattered around the electronic world.

The INFRAWEBS[1] framework is being designed and developed with the vision to help business people build composite distributed applications that tackle much of this interaction complexity as automatically as possible, with the role of human intervention not necessarily being fully dismissed. Such virtual organizations are characterized by operational and performance criteria that are independent and even sometimes incompatible with those of individual components (external services/applications) involved.

Within this global ecology of services, an INFRAWEBS application will be able to act consistently and achieve reliable performance for the user, if it manages to find stable pathways that fulfill the user-requested goal. It needs therefore to maintain an "image" of the service ecology that will represent at any moment the state of the external world with regard to the application functionality and commercial or other goals. This image or representation of the world has to be self-organizing in order to reflect the autonomous character of the external services.

B. Self-Organization in Infrawebs

INFRAWEBS applications exploit semantic web technologies to automate much of the work necessary to identify and locate web services in order to dynamically compose meaningful functionality. More precisely, applications are built and executed with the aid of a number of tools that retrieve, select, match, etc. *semantic* web services, i.e., web services whose requirements, abilities and behavior are described in one of the WSMO-supported formats (cf. Web Service Modelling Ontology website, http://www.wsmo.org/). Two use cases are being developed, one on e-tourism and one on e-government.

Within this effort, we are developing a security reasoner, which is responsible for maintaining an image of the network of external semantic web services used by an application and which follows the artificial immune systems approach.

From a design and development point of view, the usage of semantic web technologies allows the reasoner to automatically import, parse and match web service requirements (actually in WSDL format) and to offer to the system designer a straightforward possibility to select relevant "attributes" to protect.

The INFRAWEBS artificial immune system has been designed on a set of concepts that directly match those features and facilities of semantic web services. Furthermore, it uses a specialized algorithmic setup that differs from the usual setup of artificial immune systems for security.

Each semantic web service participating to an INFRAWEBS application is monitored for a set of attributes by the artificial immune system. The attributes can be either semantically described in the service description or user-defined (the user of the security reasoner is the application designer) and will be continuously given new values during service retrieval, invocation and execution. For example, in the case of e-tourism, it makes sense to use the "price" property of all airline, car rental and hotel services as a monitored or

[1] www.infrawebs.org

protected attribute. In general, global attributes can be also defined that refer to many services together. For example, in the case of e-tourism, the "average price" property of an airliner (that refers to all the individual flights it offers) is a measure of company price policy and is not expected to fluctuate as much as an individual price property of one particular flight. In principle, we can also generate new simple or composite local and global attributes that will be monitored for meaningfulness. For example, such a composite property of the type "price (Air France) > X **and** price (Iberia) > Y" could mean that price rises for the two companies are positively correlated, which could be an indication of collusion. In this case, the level of trust to these two airline companies would have to be negatively affected.

For each of the monitored attributes, an artificial antibody (protector cell) is created that monitors the attribute. Each antibody defines among its other data a target value and an activation level, which express respectively a "normal" value and the deviation from regular behavior. This setup has the following features:

- For each service and for each monitored attribute, there will always be some degree of deviation (for example, prices are never constant). Thus, the problem of detecting abnormalities and dysfunctions translates into how to detect persistent or unusual deviations, by reasoning on the history of the activation level of the antibodies.
- The normal or usual behavior of services may not be (and is usually not) known beforehand (e.g. the average normal price of a flight). Moreover, the so-called normal and usual behavior itself changes, albeit much slower than the local fluctuations. For example, average prices can slightly change (generally increase) with time, but the fluctuations around the average will be much larger than that.

These features imply that the immune network, i.e., the network of antibodies, will have to be in continuous self-organizational activity in order to approximate the, otherwise unknown, normal behavior of the service network. A network that self-organizes and reaches an equilibrium can be said to have discovered the normal behavior of the target system, i.e., of the corresponding service network. From our experience with self-organized systems, we can predict the following dynamic behavior of the immune system:

- More than one immune network can discover the normal state of the same service network.
- For this reason, some of the exact values of the various antibodies of the system may be meaningless to the application designer or may not correspond to his/her own idea about the functioning of the application.
- Continuous operation of the immune system in parallel with the corresponding service network will induce continuous change in the antibody parameters.

While these appear as drawbacks to non-informed application designers, in reality they are advantageous from a self-organization and stability point-of-view, because they allow an equilibrium to be discovered, if it exists. Continuous change in antibody parameters is the underlying activity for self-organization and occurs through two mechanisms: individual antibody adaptation and inter-antibody competition within the application [2].

C. Behavior, Performance, Measures

Measures of performance for the distributed application have to be sought that ensure the reliability of the virtual organization and its commercial or other value. To this end, we draw inspiration from biological analogs, and especially ecological networks (food webs), genetic networks, and insect socioregulatory processes.

In natural self-organizing systems, such as the above the quality of behavior "emerges" from the interactions between the components of the system. As such, it is difficult to assess computationally, let alone to predict automatically. Therefore, the quality or performance of such systems is judged on the basis of human observation in simulation ("in vitro") or, if possible, "in vivo". However, the common denominator in all natural self-organizing systems, independently of how well their macro behavior is understood, remains their remarkable ability to recover from perturbations of all kinds, to adapt temporarily to novel conditions and to develop new emergent behaviors with time as response/internalization of persistent changes in their environment. All the emergent behaviors and all the macroscopic qualities that are attributed to them, are only side-effects of this self-organization process [3].

If we want to import this approach to the design and engineering of artifacts, such as distributed computer applications, mobile networks, etc., then we should re-design these artifacts as loosely-connected systems (networks in the general sense) of ill-defined behavior. The desirable functionality would not be explicitly perceived by or described as goal of any one of the components, but it would be only visible to an external observer (human or agent at a higher level of reasoning). One of the first examples of such a system is the artificial ant-based network routing system of [6], where the optimal routes computed are not visible as a whole to any of the individual constituent agents (artificial ants), but only to an external agent, such as a human application designer.

The concept of macroscopic performance of artificial self-organizing systems as perceived by external entities may be relatively well-defined, contrary to what happens in natural systems. For example, in the above optimal routing problem, the length of the path to a destination constitutes a measure of performance. In INFRAWEBS applications, things will be more complex, because, for example, an absolute safe criterion of quality of an e-tourism plan is hard to formalize. Moreover, different users will assign different values to the same plan, so there are difficulties in assessing quality and proposing plans globally by the application.

Even more difficult, if we want a self-organizing system to develop new behaviors with time, then we should endow it with the possibility to somehow observe itself. Because by definition its constituent parts will only have a limited view of the system, then the behavioral development will have to be emergent as well and rely on individual criteria that differ from what an external observer would perceive.

Finally, our experience with natural self-organizing systems implies that their behavior under stable conditions is generally not optimal because of continuous intrinsic self-organization not triggered by external stress. This feature is the price to pay for the potential of recovery from perturbations, resistance to stress and behavioral development.

As a result of all the above, the first step to successful design of performance measures for a self-organizing system, is the successful design of the system itself as a system demonstrating emergence instead of optimality. The subsequent design of specific measures will have to comply with the ill-defined nature of self-organizing systems and would better be based on generic concepts of self-organization appropriately instantiated.

One characteristic set of such concepts is the quadruple (*stability*, *resistance to perturbations*, *predictability*, *development*). We proceed below to describe in brief how this approach is treated within INFRAWEBS application development.

D. Design of Measures

The most evident manifestation of ***stability*** in a self-organizing system is the existence of an equilibrium. Because this will be emergent, we cannot in general predict in advance which measure will reach an equilibrium. Thus, we should discover this measure during online trials at design time. One approach compatible with the overall line of thought is the selectionist one, according to which many measures can be *generated* and tried for equilibria, and the measures corresponding to the best equilibria selected for adoption.

In INFRAWEBS applications, individual antibodies used should be more-or-less in equilibrium, just like their biological counterparts, and the same should apply to generic population measures (in our case we have used the "average activation level" measure). For example, in an e-commerce application, there exist antibodies for the average price exposed by each service and/or for the average total price. Although we don't expect *any* price to be constant in time, it is reasonable to assume that a constant average price would be one safe criterion of a stable organization, i.e., of one organization in ecological equilibrium. Thus average prices in this case can be used as measures of stability.

As a rule, and from previous experience with biological self-organizing systems, we can expect many global measures to be in equilibrium, even if they don't directly represent real functionalities in the virtual organization, for example the number of services *changing* their exposed price. This is an advantage to us designers, because we can derive many stability criteria for the same organization. This can be also a disadvantage, because the observed variables may not be meaningful from the application's point of view. One design guideline for equilibria measure design is to search ***behavioral attractors*** of the system, i.e., attractors for derived values and measures that express the functionality and operational goals of the system (such as average price), rather than purely structural parameters (such as connectivity metrics).

The second concept of ***resistance to perturbations*** can also be instantiated by systematic trial and error. For example, in an e-commerce application, if suddenly the number of available services for one task doubles, this constitutes a perturbation. If the overall system can regain stability after a while, this is an indication of an operational, "healthy" application. In such cases, the matter of study is the relation between the former and the new equilibria. The analysis of the perturbations goes in two directions: (i) we don't expect *any* perturbation to allow the system to find a new equilibrium, and (ii) sometimes the new equilibrium is unacceptable for the organization.

Thus we need to find ways to ***predict*** system behavior in presence of a number of perturbations and this can be accomplished in off-line ***simulation***. For example, in the previous e-commerce case, we can simulate the system to find the threshold above which the number of newly inserted services will be unmanageable by the system (will not lead to an equilibrium or will lead to an unacceptable equilibrium). The study can be thought to be over when (i) the behavior of the system can be at least qualitatively predicted within a range of operation, and (ii) the major perturbations considered can be accomodated by the system within acceptable limits (for example, if the number of new services is up to a quarter of the existing services).

We also need to devise and study counter-measures to specific perturbations, i.e., ***counter-perturbations*** that can make up for potentially harmful ones. For example, in the above case we can think of introducing virtual services of other types, or one virtual service that will act as an interface (filter) between the organization and the potentially umlimited number of external services of the critical type.

Lessons from highly perturbed biological systems can be imported, for example from host-parasite ecologies, ecologies with an externally manipulated food source, etc.

The most important aspect of such biologically-inspired equilibria remains however the dynamic nature of the equilibria themselves. For example, in e-commerce applications, we know from everyday experience that, in the long term, average prices are not constant but change as a result of many factors, including inflation, technological upgrades, competition, etc. A virtual organization should be able to maintain by self-organization a correct image of the ***developing equilibrium***, in the same way that a real biological system learns from experience. Although the study of developing, dynamic equilibria depends directly on real data of external world dynamics, we can predict one mechanism that allows this kind of learning. Namely, a second organization, parallel with the first and of the same nature, can reason on medium-term parameters or measures of the organization (and not directly on the state of external services themselves) and self-organize accordingly. ***Meta-reasoning*** of this type can act as a self-regulatory process that allows the organization to maintain a dynamic representation of the global world, much in the way that an insect society regulates the allocation of its work force to different social tasks.

In sum, the qualitative concepts of stability, resistance to perturbations, predictability and development, which are well-defined within self-organization research, have to be instantiated for a particular self-organized system in, possibly many, measures, sometimes overlapping or contradictory. In the case of INFRAWEBS applications, the measures can be generated and the corresponding data collected semi-automatically with only little additional effort, thanks to the use of semantic technologies to describe the capabilities, goals and behaviors of the various services involved.

E. Perspectives

The study of performance and the definition of measures for ill-defined systems is an open research theme. Previous discussions [4][5] in the domain of intelligent systems that are hard to qualify concluded as well that performance should be qualitatively assessed to fit the ill-defined nature of the domain. In this respect, self-organized systems are much worse than intelligent systems, because they are by definition distributed, mostly large-scale and they exhibit emergent behavior. In another line of research, the formal attributes of particular self-organized systems are studied, outside any application realm, for example see [1][7] for artificial immune systems.

As far as INFRAWEBS applications are concerned, the adopted direction of work allows in the middle term rapid development and prototyping. Farther in this direction, more advanced issues will come into play: incorporation of potentially harmful features (such as viruses) in the system after persistent external manipulation, stabilization to suboptimal equilibria (from an application perspective), cyclic equilibria, etc. In the longer term we can think of cancerous organizations, threats spread, deception of one service by another, etc.

References

[1] Garrett, S., “How do we evaluate artificial immune systems?”, *Evolutionary Computation*, 13(2):145-178, 2005.

[2] INFRAWEBS Deliverable 8.3.2, Design security and privacy models and application, July 2006.

[3] Kauffman, S.A. *The origins of order - Self-organization and selection in evolution*, Oxford University Press, 1993.

[4] Panel discussion “Performance Metrics for Intelligent Systems” (IEEE *Intelligent Control Systems Conference 2000*, Patras, Greece, July 2000).

[5] *Proceedings of the Workshop on Performance Metrics for Intelligent Systems*, Washington, DC, August 2000, edited by A. Meystel.

[6] Schoonderwoerd, R. Holland, O.E., Bruten, J.L., Rothkrantz, L., “Ant-Based Load Balancing in Telecommunications Networks”, *Adaptive Behavior*, 5(2): 169-207, 1996.

[7] Network for Artificial Immune Systems (funded by EPSRC), http://www.elec.york.ac.uk/ARTIST/.