Simple Ecological Rules Yield Complex Agent Networks

Miguel LURGI a,b,1 and David ROBERTSON a

a School of Informatics, University of Edinburgh, Scotland, UK.
b MicroArt S.L., Barcelona, Spain.

Abstract. In complex natural systems (such as ecological systems) it is common for those studying the macro-dynamics of systems to aim to describe, as simply as possible, the interactions between individuals occupying different niches in a system and to validate these by studying whether they account for the complex network of relationships observed in the actual system. We explore a similar principle applied to multi-agent systems: we develop interactions among intelligent agents; these interactions produce complex networks of relationships among the entities in the system in a similar fashion in which ecological networks are formed in nature. The study of the topological features displayed by these networks is helpful for determining the stability of the system of software agents and its relation to the complexity of the network itself. An interaction centred approach for coordination and knowledge sharing among artificial agents is adopted for the implementation of the system.

Keywords. multi-agent systems, ecology, complex networks, collaboration

Introduction

Charles Darwin described the intricate network of relationships between species in a natural community using the metaphor of an “entangled bank”, in which he depicted a typical natural scene of a group of different species living together and interacting; in this example scene we can see represented the web of complex relations by which species in natural ecosystems are bound together [1].

A fruitful approach that has been widely adopted in the realm of ecology for the study, and better understanding, of this complex set of entangled interactions is that of ecological networks. In ecological networks, the analyses of the relations between species, which are represented in the form of a graph, are performed using tools borrowed from network theory. These analyses have generally found interesting structural patterns within these networks, which are believed to account for key system-level features like their stability and persistence.

When engineering multi-agent based systems it is useful for our agents to interact in such ways that these interactions provide the system with the features of stability and persistence found in natural systems as described above. We are interested then in pro-

---

1Corresponding Author: Miguel Lurgi, School of Informatics, University of Edinburgh, Scotland, UK; E-mail: miguel.lurgi@ed.ac.uk
viding our intelligent agents with the necessary properties to create a network of interactions displaying the structural patterns commonly found in networks of relationships between natural species.

From this perspective, we are interested in addressing particular questions such as:

1. is it possible to engineer multi-agent systems in such a way that the networks of interactions emerging among the agents within them display the structure and patterns encountered in complex natural networks?,
2. can we adopt the processes describing interactions among individuals in natural communities, and which are believed to account for the patterns found at the ecosystem level, to specify interactions between agents?,
3. will these representations of natural processes yield complex networks of interactions amongst our digital entities as they do in nature?,
4. and can we find patterns in the networks of relationships between our digital agents that will permit us to draw interesting conclusions about properties of the system such as its stability and/or persistence?

To address these questions, we take ecology as inspiration and propose a method for describing interactions between artificial agents within a multi-agent system, based on simple ecological rules, and which are grounded on the processes that are believed to be responsible for the structure of networks of interactions in ecological communities [2].

We adopt an interaction centred approach for agents coordination and knowledge sharing for the implementation of the multi-agent system subject of this work.

1. An Interaction Centred Approach for Agent Systems

The field of multi-agent systems is an active and ever increasing area of research, and many different approaches have been proposed for the representation of intelligence, communication, and collaboration within these distributed systems [3, 4]. The evolution of agency has seen a steady development in the last decade, and recently, approaches based on the specification of protocols of interaction for agent coordination and communication have been demonstrated to be useful in large systems because of their scalability and compactness [5].

One of the major advantages of these approaches is the fact that no common ontologies need to be shared in advance at the system level. Agents need only to find an appropriate interaction protocol and other agents taking part in it. Ontological alignment then is performed at the time of the interaction and only between the partners involved in it, encapsulating in this way, at the interaction level and time, part of the complexity of multi-agent interaction.

1.1. Lightweight Coordination Calculus and OpenKnowledge

For addressing the problem of interoperability and cooperation in multi-agent systems, some efforts have been made towards the design of languages for the standardisation of communications between the systems’ components, like for example through the creation of preformative-based message passing protocols such as FIPA-ACL [6] and KQML [7]. The aspiration of these languages is to maintain basic rules of social inter-
action when groups of agents are working together in a particular task, while allowing at the same time sufficient autonomy for each individual agent.

The Lightweight Coordination Calculus (LCC) [10] was proposed as a process calculus for the specification of electronic institutions [8, 9] in multi-agent systems operating in peer-to-peer environments. We are interested in using LCC for the definition of the interactions among the agents in our digital systems because it offers a compact method for specifying social norms in societies of agents that permits simple and powerful mechanisms for analysis and deployment.

For the implementation of a multi-agent system based on protocols for the formalisation and description of the interactions among the entities sharing our digital environment, we use the OpenKnowledge system [11, 12]; a peer-to-peer (P2P) based system that makes use of the LCC for the specification of the interaction models among autonomous agents which communicate through a P2P network.

The flexibility provided by the OpenKnowledge system in terms of authoring, sharing and discovery of interaction models and role implementations (components) gives our ecological interactions the added value of being easily discoverable and adopted by other peers in the OpenKnowledge network, promoting in this way the formation of a digital ecosystem of interacting partners.

2. Simple Ecological Rules

The rules for interaction between the intelligent agents in a multi-agent system can be defined following different patterns and methodologies. In this particular case, we have found that basic ecological rules, of the kind that are often used to describe species interactions in natural systems, have the potential of creating complex networks of interacting artificial entities.

We are mainly interested in agents capable of collaboration and cooperation, in which both partners of any given interaction will benefit from its outcomes. For this reason we have defined the ecological rules inspiring the interactions between our agents based on a particular kind of interaction that is found in nature, and which in ecology is known as mutualism, i.e. a kind of interaction in which both of the species involved benefit from it.

2.1. Mutualistic Networks

The research area of ecological networks is concerned with the analysis of the relationships amongst species in an ecosystem from a network perspective, which has allowed scientists to draw interesting conclusions about the complexity and stability of natural communities [13].

A particular type of ecological networks that have been the focus of attention recently, are those representing mutualistic interactions amongst species (i.e. mutualistic networks). Research on this kind of interactions networks has demonstrated that their structure minimises competition and confers species diversity and stability to the system, either in natural [14] and artificial [15] ones.

Taking inspiration from mutualisms and the processes undermining their emergence, we define the protocols of interactions and the behaviour of our artificial agents based on
the way species in mutualistic networks interact; with the ultimate goal of providing the ecological groundings for cooperation and diversity of entities to emerge in the network of interactions amongst the infohabitants of our digital environment.

The definition of our agents and the protocols through which they interact with each other are thus based on a set of ecological concepts:

1. **Traits**: a set of characteristics that define any entity within the system.
2. **Degree of complementarity**: the degree to which any trait of a given infohabitant is complementary to another trait possessed by another infohabitant.
3. **Habitat**: a representation of the part of the environment where the agent spends most of its live. An agent is more propense to interact with others in its same habitat.
4. **Niche**: the multidimensional space composed by different characteristics of the environment (relationships) in which a given agent lives.
5. **Meta-communities**: aggregation of entities that belong to different habitats (or regions) and that occasionally interact when certain conditions are met.
6. **Fitness**: a measure that is used to determine how good an entity is doing during its lifetime in the digital ecosystem.

This is an example, written in the LCC language, of the rules (protocol), based on the ecological processes described, that our agents follow when interacting in this environment:

```
01 a(visitor, X)::
02 null <- chooseHost(Y, ListOfHosts)
03 then
04 whereabout => a(host, Y)
05 then
06 ( in(HabitatHost) <= a(host, Y)
07 then
08 ( ( ( null <- sameHabitat(HabitatHost)
09 then
10 whichtrait => a(host, Y))
11 or
12 ( null <- metaCommunity()
13 then
14 whichtrait => a(host, Y)))
15 then
16 availabletrait(Trait) <= a(host, Y)
17 then
18 null <- haveTrait(Trait)
19 then
20 whichsize => a(host, Y)
21 then
22 size(TraitSize) <= a(host, Y)
23 then
24 ( null <- complementary(TraitSize)
25 then
26 null <- need(Amount, Reward)
27 then
28 exchange(Amount, Reward) => a(host, Y)
29 then
30 offer(Offered) <= a(host, Y)
31 then
32 null <- consumeResource(Offered, Reward)))
```
or
( quit => a(host, Y)))
then
a(visitor, X)

a(host, Y)::
whereabout <= a(visitor, X)
then
( null <- location(Habitat)
then
in(Habitat) => a(visitor, X)
then
( ( quit <= a(visitor, X))
or
( whichtrait <= a(visitor, X)
then
null <- myTrait(Trait)
then
availabletrait(Trait) => a(visitor, X)
then
( ( quit <= a(visitor, X))
or
( whichsize <= a(visitor, X)
then
null <- traitSize(TraitSize)
then
size(TraitSize) => a(visitor, X)
then
( ( quit <= a(visitor, X))
or
( exchange(A, R) <= a(visitor, X)
then
null <- obtained(R)
then
null <- has(A, O)
then
offer(O) => a(visitor, X)
then
null <= synthesis(R, O))))
then
a(host, Y) ) )

Where null denotes an event which does not involve message passing; the operator :: is used to declare the definition of a role within the protocol; and the operators <=, then and or are connectives for logical implication, sequence and choice respectively. M => A denotes that a message, M, is sent out to agent A. M <= A denotes that a message, M, from agent A is received.

In our example we have defined the roles of “host” and “visitor” in resemblance to the actors taking part in an ecological mutualistic interaction. Leaving some of the details aside, we can see how the ecological concepts introduced above of: traits, degree of complementarity, habitat, and meta-communities; are explicitly introduced within the interaction protocol.

In any given execution flow of this protocol the agent acting as visitor will initiate the interaction by searching for appropriate partners (line 2 in the protocol); for this action the chooseHost predicate is used for selecting one of the available hosts in ListOfHosts. The selected host is referenced by the variable Y. Once it finds a suitable host, the inter-
action develops through the exchange of a series of messages where the agents exchange information about:

1. The habitat they are inhabiting and whether they will form a meta-community (lines 4-8, 12 and 38-42). In this part of the interaction the visitor agent sends the *whereabout* message to the host in order to determine its habitat location, and consequently, the latter responds using the *in* message which contains a reference to the habitat (*Habitat, HabitatHost*) in which it is located. The host obtains a reference to its habitat employing the *location* constraint. The visitor agent then checks whether it shares habitat with the host using the *sameHabitat* predicate; if they do not share the same habitat, the visitor agent can decide whether to form a meta-community with the host agent using the *metaCommunity* constraint.

2. The matching trait for the interaction (lines 10, 14-18 and 46-50). This step is performed using the *whichtrait* message, sent by the visitor agent to the host in order to find out the trait of interest. The host solves the constraint *myTrait*, and sends its *Trait* to the visitor through the *availableTrait* message. The visitor then confirms whether it can cope with that *Trait* using the *haveTrait* predicate.

3. Its degree of complementarity (lines 20-24 and 54-58). The information about the degree to which the matching trait is complementary is exchanged using the *whichsize* and *size* messages. The host uses the *traitSize* constraint to determine a value (*TraitSize*) used to evaluate the trait compatibility, which is finally evaluated by the visitor through the *complementary* constraint.

4. And the resource amount they will exchange (lines 26-32 and 62-70). Here the messages *exchange* and *offer* are employed to exchange information about the *Amount* of resource required by the visitor and the *Reward* offered to the host. The host assimilates the reward (*R*) using the *obtained* constraint and determines the amount of resource (*O*) it is able to offer through the *has* predicate. Finally, the exchanged resources are consumed using the *consumeResource* and *synthesis* constraints by the visitor and host agents respectively.

The “*quit*” message (lines 34, 44, 52, and 60 in the protocol) is used by the agents to terminate the interaction whenever is found that the necessary ecological conditions for it to happen are not met.

The concepts of niche and fitness are embedded within the agents themselves and represent the protocols available for the each agent on the system, and a measure of how well any given agent is doing, respectively.

### 3. Complex Agent Networks

We aim to employ protocols of interactions, such as the one introduced above, for enabling collaboration between entities within a digital ecosystem [16]. In order to be able to collaborate, agents in a digital ecosystem need to gather information about the capabilities, and degree of each of these capabilities, of other agents in the system. This information exchange will enable them to find suitable partners for interaction, depending not only on the capabilities of others (services they can provide) but also on the needs of the agent itself. Apart from this, agents in this kind of systems also need to discern how much a particular service provided by another agent is going to cost and what are the potential benefits obtained from it. These are the bases of collaboration.
In particular scenarios like the one just described, in which agents might want to exchange information about their “traits”, the degree to which those are complementary (offered vs needed), and the resources they might be willing to exchange; a protocol of interaction like the one presented in the previous section is a good representation of the type of interactions that agents in these systems might embark upon.

For evaluating the extent to which the ecological protocol presented is an effective way of promoting the creation of robust networks of agents, we performed a series of 20 simulation runs with 25 agents interacting in a digital environment enabled by the OpenKnowledge platform and following the described protocol. As stated above, the protocol requires for its execution the presence of agents assuming two different roles: host and visitor. In our simulations the agents were only prepared for taking one of the roles available, although they could take both of the roles if necessary. This was done in this way to facilitate the interpretation of the resulting network of interactions and to allow for a direct comparison with the ecological networks focus of our attention: mutualistic networks, which are represented in the form of a bipartite graph (see [17]), since there are not intra-specific relations amongst entities of the same type (roles in our case).

The initial setting for the simulation included the initialisation of the parameters by which each agent will interact with the others: the habitat it lives in, the traits it will use for interactions, and an integer value associated to any given trait that is used to compare the extent to which the agents are complementary on their traits; the amount of resource that each agent has available for exchanging is implemented as a stochastic function partially dependant on the resources available and the resources obtained/offered by the other agent taking part in the interaction. These parameters were initialised in such a way to produce an heterogeneous community of agents. Each agent initially possesses only information related to its own necessities and capabilities, information needed from other agents is acquired at the interaction time and only through the messages allowed by the interaction protocol.

The simulation runs were allowed to evolve for a fixed number of successfully completed interactions: five hundred (500). This number was selected based on the observation of previous runs in which the network of interactions reached an stable state, in the sense that its topological configuration was not noticeably affected by further interactions, as the number of interactions amongst the agents in the system approached this number.

Relationships among pairs of agents arose with different strengths (the number of times an agent interacts with any other relative to the number of times it has interacted during the entire simulation) and with different configurations.

Although a quantitative analysis for the comparison and evaluation of the runs carried out remains to be done, a preliminary qualitative analysis of the resulting networks of interactions from each of the runs was performed. We compared the obtained networks from all of the runs and apart from the identity of the agents, which is changed in every run based on the different configurations they can be initialised with (as explained above), the network architecture was, in general, similar among the obtained interactions networks; presenting constant patterns that are also found in other types of networks, as described below.

Figure 1 shows an example, taken from one of the runs in our experiments, where these relationships are represented in a fashion similar to networks of interacting species.
Figure 1. An example of the network of relationships emerging from a collection of software agents allowed to interact through an interaction protocol based on simple ecological rules. The open nodes represent the agents taking the "host" role, and the grey nodes the agents with the "visitor" role (see text).

described in the field of ecology (as we saw in section 2). In this graph, the agents and their relationships with other agents are represented by nodes and arcs respectively. A link (arc) is generated between two agents whenever an interaction is successfully completed amongst them. It is worth mentioning at this point that in the field of ecology the relationships amongst species when represented in the form of a graph tend to be directed, with the direction of the arc between two species representing the direction of the flow of biomass/energy between them (from host to visitor); it is for this reason that our graphs are displayed in this way.

By representing the relationships between the agents in our digital ecosystem in this way we are able to extract features, obtain descriptors, and perform analyses over the resulting network based on methods borrowed from network theory. We obtained networks that display a scale-free structure: the majority of nodes have small degree ($\leq 2$), while a low fraction of them are highly connected (as can be seen in the figure); and small-world properties: with short paths between any two nodes; properties which are common patterns found in different kinds of complex networks in nature and the artificial world [18], and which differ significantly from the structure that we would expect from a randomly assembled network.

Another property seen in our networks, which is related to their scale-free character, is the preferential attachment displayed by visitor agents with low degrees (the two grey nodes on the top right corner of figure 1) to host agents that are highly connected. This is a common feature encountered in mutualistic networks, where specialist species are more likely to interact with generalists [17]. Asymmetric specialization (ie, a specialist interacting with a generalist) has been found to be a pervasive feature of plant-pollinator interactions networks, and it is believed to be beneficial for the majority of species because it facilitates the avoidance of extinction risks when species are highly reciprocally specialised [19].
These patterns are important in practice because, as it has been argued (see [18] and references therein), they can give us information about functional properties of the system such as: information propagation speed and resistance to node failures, which in turn provide us with a better understanding of the relationship between the complexity and stability of our agents systems. We can employ this information to carefully analyse agents that play important roles in our digital ecosystems (e.g. agents acting as hubs of information) to understand what are the features that set them apart from other agents.

The patterns encountered in the networks of relationships amongst our artificial agents are in many ways similar to those found in ecological mutualistic networks, as shown above; which are patterns that differentiate random networks from self-organised complex networks of relationships. This network architecture is an emergent property of our system since the only mechanisms involved in agents’ interactions are those specified by the protocol of interaction presented above. The creation of such a complex and intricate pattern of relationships is not an implicit property of the simulations performed but rather the product of many different agents interacting together for achieving their respective goals (gather resources and survive).

By comparing the structure of the networks of relationships resulting from our simulations, with the networks of interactions amongst mutualistic species in nature, we can translate the results of the research performed until now in the field of ecology in general, and ecological networks in particular (literature cited so far), into the domain of multi-agent systems. The characteristics described above, which are shared between our networks of artificial agents and those observed between species in the natural world, let us conclude then that the features conferred to these natural networks by the characteristics they display are also given to our agent systems thanks to the emergence of a similar network structure to that found in these ecological networks.

4. Discussion

We were able to engineer a system capable of displaying networks of interactions among agents that are similar in structure and patterns to those found in natural ecosystems, as shown in the previous section. This was the focus of attention of the first question proposed in the introduction.

We have given an answer to question 2 by using simple ecological rules, inspired by processes that describe interactions among species in real communities, for the design and development of interactions among autonomous agents in a multi-agent based system. This approach has allowed us to obtain complex networks of interactions among our digital entities that possess similar features to those that characterise complex networks in real communities and that are believed to be responsible for their stability and persistence; this answers our third question.

The structural patterns encountered in our interactions networks can give us useful insights about important properties of our system of interacting agents that allow for a better understanding of its dynamics, answering in this way our fourth question.

We are now focused in studying this kind of networks in a more formal way from a network theory perspective, in order to quantify the exact similitudes between the obtained artificial networks and the actual networks occurring in natural communities. As in real ecological networks, these analyses can provide us with important insights about the stability and other important features of our networks of agents (e.g. [20]).
Acknowledgements

This work is part of the project “EcoBusiness: A Multi-Agent Approach for Digital Business Ecosystems”, funded by the European Commission through the Seventh Framework programme Marie Curie Actions - Industry-Academia Partnerships and Pathways (IAPP). Grant agreement no.: 230618.

References