SIMULATION
http://sim.sagepub.com/

Model resolution in complex systems simulation: Agent preferences, behavior, dynamics and *n*-tiered networks

Eugene Ch'ng

SIMULATION 2013 89: 635 originally published online 5 March 2013
DOI: 10.1177/0037549712470582

The online version of this article can be found at:
http://sim.sagepub.com/content/89/5/635

Published by:
SAGE
http://www.sagepublications.com

On behalf of:

Society for Modeling and Simulation International (SCS)

Additional services and information for *SIMULATION* can be found at:

Email Alerts: http://sim.sagepub.com/cgi/alerts

Subscriptions: http://sim.sagepub.com/subscriptions

Reprints: http://www.sagepub.com/journalsReprints.nav

Permissions: http://www.sagepub.com/journalsPermissions.nav

Citations: http://sim.sagepub.com/content/89/5/635.refs.html

>> Version of Record - Jun 12, 2013

OnlineFirst Version of Record - Mar 5, 2013

What is This?

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

Simulation

Simulation: Transactions of the Society for Modeling and Simulation International
89(5) 635–659
© 2013 The Society for Modeling and Simulation International
DOI: 10.1177/0037549712470582
sim.sagepub.com
SAGE SCS

Model resolution in complex systems simulation: Agent preferences, behavior, dynamics and *n*-tiered networks

Eugene Ch'ng

Abstract

Agent-based modeling is a process of representing and simulating the intentions, behaviors and actions of complex systems with the goal of understanding specific phenomena related to the communications within complex systems that produce emergent behavior and self-organization, or for predicting spatial or behavioral patterns of individuals or groups of interacting entities. Agent-based modeling, also termed multi-agent systems, or in ecological simulation, individual-based models, spans simple to highly complex systems; their interactions can be difficult to implement and optimize programmatically, particularly when there could be hundreds of thousands of agents within a community that have multiple levels of communication. The resolution and the scale of simulation is an especially important component that could determine the accuracy of the models. This article focuses on the model resolution of complex systems, facilitated by an object-oriented communications framework, a foundation for the simulation of the fine resolution of the dynamics, behavior, preferences, interaction and *n*-tiered trophic networks, including the simulated environments they inhabit. It dissects individual agents with a view to modeling and simulating fine behaviors amongst a population of agent types in *n*-tiered networks, scalable to hundreds of thousands of species using mathematically defined behavior, efficient algorithms and adaptive data structures as support for the simulations.

Keywords

simulation system architecture, modeling and simulation environments, agent-based systems, complexity, artificial life

1. Introduction

Complexity science studies systems that have a population of interacting entities. Theories in complexity complement classical mechanistic perceptions with the aim of explaining the broader phenomenon of a system or systems. Following the reductionist approach of René Descartes that the complexity of the world can be understood by reducing it to the interaction of parts or simpler things, the notion given by Sir Isaac Newton that ''Truth is ever to be found in simplicity, and not in the multiplicity and confusion of things''[1] and the idea of a ''clockwork universe'' has influenced past scientific thinking. It was understood that once you have analyzed phenomenon in their simplest components, ''their evolution will turn out to be perfectly regular, reversible and predictable, while the knowledge you gained will merely be a reflection of that pre-existing order''. As scientific explorations expanded into the study of living organisms and populations in different orders, however, the mechanistic conception has reached a limit in its ability to yield new knowledge, as far as complex systems are concerned.

''Complex'' describes a system as consisting of many different and connected parts; it is a network, a group or a system of different entities that are linked in a close or complicated way – according to Oxford Dictionaries.[2] Complex adaptive systems are present in every level of the hierarchy of life – at the molecular level to individual organisms and metapopulation, and from community to the global environment. The intricate relationships at the micro level and the emergent behavior found at the meso and the macro levels cannot be understood by analyzing the micro states alone, for ''the whole is more than the

IBM Visual and Spatial Technology Centre, University of Birmingham, UK

Corresponding author:
Eugene Ch'ng, IBM Visual and Spatial Technology Centre, University of Birmingham, Edgbaston B15 2TT, UK.
Email: e.chng@bham.ac.uk

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

sum of its parts''.[3] Complex systems require both a holistic and a reductionist approach in the epistemology; the latter is needed at least in the modeling, in order to understand the phenomena that are manifested in various levels of organization. Although the variation of complex systems is vast, they share common characteristics, such as emergence and self-organization,[4] that resulted from the interaction of simple rules. Complex systems are found at 'the edge of chaos':[5] they exist between order and disorder, they are predictable and at times unpredictable; miniscule changes in initial conditions produces massive stochastic revolutions at spatial and temporal scales. These features may give us a ''handle'' from which we can grasp illusive behavior at the systemic level for creating models and for understanding phenomenon. How local communication between agents gives rise to collective behavior is an important study area. At present, however, the inner workings of many complex systems are yet unknown. Otherwise, issues related to epidemics, medical treatment, society, the economy, ecology, etc. would have been resolved. Agent-based modeling and simulation can provide a platform from which phenomena can be understood within our lifetime, at a flexible spatial and temporal magnitude, at liberty from ethical and environmental constraints, and free from noise. We must not forget that a model is a model; however, new information can be acquired and conclusions can be made from hypothesis generation and testing. The foundation of understanding complex systems requires computer modeling, for we are dealing with non-linear interactions within a population of entities. These interactions are mainly information and energy communications. They are difficult to model and predict with state-variable techniques (difference and differential equations) as changes occur at different levels and at certain resolutions.

Modeling complex systems necessarily implies that we are interested in determining the properties, functions and principles associated with an individual entity, and how that entity interacts and communicates information with other entities and its environment. We must not, however, observe behaviors at a local level only. The collective behavior at multiple levels (micro-meso-macro) will need to be analyzed. This is what is meant by the need for both a holistic and a reductionist approach. Such a modeling procedure is bottom-up, reflecting how complex systems in nature work.[6] This can be contrasted with ''top-down approaches'', where models are governed by a central system that plans, arranges and tunes patterns (see Ch'ng[7] for a survey of traditional statistical and predictive modeling methods).

In particular, complex systems modeling and simulation remains a challenge in many ways. These challenges can be grouped into three categories. Firstly, the resolution of the model must allow for individuals to be represented as individuals and not as groups of individuals, although certain entities aggregate as an organism (e.g., the Slime Mold[8]). The model should, at the lowest level, support individuals. These individuals could be as small as unicellular organisms. The spatial and temporal resolution must also be at a scale in which realistic movements at a level of continuity, as opposed to coarse discrete steps, are supported. The resolution of agent behavior and dynamics must also reflect the resolution of its mobility within time and space. The resolution of time, space and the context of the space at which the agents dwell therefore are important factors determining the formation of complex adaptive networks. Models must allow agent movements within a space, not just an abstract model. Secondly, there are multiple levels of networks spanning millions of agents; the way in which they can be represented as object-oriented (OO) classes and the cycles of interaction and communication can be difficult to manage programmatically. The computational complexity rises immensely in ecological system simulation. The third issue relates to the optimized OO framework that facilitates efficiency and scalability with agent–agent and agent–environment interaction and communication management that make use of features in object orientation, such as inheritance and polymorphism, within an adaptable structure. Taking an object-orientation approach also means that a community of hundreds to thousands of species is manageable programmatically. The second and third issues are closely related during design and implementation. Current agent-based model (ABM) simulators do not support the emergence of these functionalities, but developers will quickly discover that the experience of many of these systems leads to the desire to implement their own, for the sake of flexibility, scalability and control over how information flows within the framework. Many of these systems are not built with a wider purpose in mind.

This article attempts to establish the resolution of the model – the dynamics, behavior, preferences, interaction and *n*-tiered networks in complex systems, including the simulated environments they inhabit, within a flexible framework for large-scale and possibly exa-scale simulation when parallel and distributed algorithms related to the framework are developed. Examples of such use can be for the modeling of prehistoric and future climate change spanning thousands of years, ecology, biodiversity informatics, societal phenomenon, cultural shifts and information flow within economic entities. This paper dissects individual agents and complex adaptive systems as a whole with a view to modeling and simulating fine behavior amongst small numbers of agent types with potentials to scale up to hundreds of thousands of agents later using mathematically defined behavior, efficient algorithms and adaptive data structures as support for simulation. All complex systems have similarities – they are composed of individuals, they have preferences, behaviors (simple rules), they interact, the individuals communicate information and

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

energy, and they inhabit an environment with micro and macro conditions. These phenomena allow us to generalize their features for modeling. The model presented here focuses on biological and ecological systems as the diversity of their characteristics encompass all other systems (e.g., social, economics, etc.). The model, in this case, is applicable to a wide variety of fields where complex systems are studied.

2. Challenges in complex systems modeling and simulation

There are challenges in complex systems modeling and simulation. How can virtual environments incorporate global and local climate, environmental conditions, etc., at a resolution that allow agents to react, contribute to and develop in a way that is similar to the real world? Can we redefine the rules of interaction so that complex rules in nature can be simplified in a way that computing resources are used at a minimum for each agent? What are the common dynamics, biological, behavioral and social patterns found in the different aspects of simple and complex organisms that could be modeled in OO concepts? Trophic networks, resource usage, social relationships and environmental factors are intertwined in the real world. How can we programmatically administer agent network interactions and communications in an efficient way? These questions are associated with the focus of the paper laid out in the introduction, and are covered in the following sections.

2.1 Simulation Environment

Simple environments afford simple agents, while complex environments create diversity. Applying Herbert Simon's Ant[9] to the modeling context, complex environments facilitates complex behaviors from agents with simple rules. Past research related to a relatively large population of agent-based modeling has looked into environmental factors such as temperature (local, global and altitudinal), sunlight, humidity, level of carbon dioxide in the atmosphere and soil types (depth, acidity, texture, slope and nutrients).[7,10–12] The models, however, are mainly global with implicit local conditions, for example, clustering of vegetation causes soil nutrients to increase due to decay of plants; clustering of canopies changes the local temperature and shade. Although the models successfully simulated micro to macro to micro effects (explained in detail in Section 2.2), for example, the clustering of individual vegetation (micro) forms a state of change in the temperature/shade in that cluster (meso), which in turn affects the overall macro-level property of the system and, as a consequence, affects the micro state of the environment, more explicit environmental parameters need to be defined for greater control of other local environmental factors. The explicit local condition of the environment requires new types of modeling. Therefore, a balance of simple rules and complex interactions must be investigated.

2.2 Modeling the individual

The application of complexity theory to the modeling of complex systems requires that we depart temporarily from state-variable models. The new paradigm of looking at the world and of modeling physical agents (biological and social) is also called Individual-based Modeling (IBM), Individual-based Ecology (IBE)[13] or Multi-agent Systems (MASs). IBE was seen by Huston et al.[14] in a visionary article as a potential modeling approach that could unify ecological theory. The approach is capable of simulating the effects of individual variation, spatial processes, cumulative stress and natural complexities that are difficult or impossible to address with classical approaches, such as state-variable models. State-variable models, whilst useful at a later stage for analyzing and projecting results, are limited as they operate at the population level and represent population as an overall state,[15] therefore disregarding natural complexities associated with individual interactions in differing spatial and temporal dimensions. Regarding predictive modeling, a number of critiques have recently questioned the validity of such modeling strategies; Pearson and Dawson[16] showed that many factors other than climate determine species distributions and, while predictive modeling focuses on the identification of a species bioclimatic envelope, other factors such as biotic interactions, evolutionary change and species dispersals are not taken into account. Furthermore, factors affecting species distribution at the local level, such as natural barriers and local climate, are not taken into account in many of these models. The majority, if not all, of Grim and Railsback's reviews related to IBM or IBE[13,17] in the past decade are discrete cell-based environments; continuous environments remain a challenge in the field, although research has been conducted in both sessile and vagile organisms.[10,11,18–22]

Modeling agents implies taking an interest in the properties and functions associated with an individual entity and how that entity interacts with other entities and its environment to produce a macro state that affects the individuals at the micro-level.

The 'micro' and the 'macro' were viewed as dichotomies in sociological theorizing until attempts at integrating the two in the 1990s and before:[23] this is termed the micro–macro link. The micro–macro link is a problem mentioned initially in sociology and the term was used in more articles in sociology (e.g., Sawyer;[24] Schillo et al.;[25] Drogoul and Ferber;[26] Buskens et al.;[27] Squazzoni;[28] Conte and Castelfranchi[29]) than in other fields, although the issue has also been used to understand ecology,[30,31]

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

settlement and land use,[32–34] economy[35–37] and culture[38] as far as simulations and MASs are concerned.

Schillo et al.[25] identified the four different perspectives of observation in the micro–macro link: micro level, meso level, macro level and meta level in sociology. Schillo et al. categorized the micro level as related to inter-human behavior, the meso level as related to institutions and organizations, the macro level as related to the whole society, and the meta level as related to ideas about society and as criticism of ideology. The relativity of describing where the level stands can be different in the hierarchy of complex systems. As Gernstein's[35] second principle (p.90) suggests, ''macro and micro are general, mutually correlative terms, which may be usefully instantiated in many different ways''. An example will make this clearer – a cell may be at a micro level as compared with an organ, whereas an animal maybe at a micro level as compared to a community.

The micro and macro levels are intimately linked in different complex systems, although the link may be inarticulate and may not be cyclical. There is certainly a link between the macro-structural phenomena of a system and its causal influence on individuals,[24] either constraining or coordinating them.[39] For example, Nowak and Latané's[40] social simulation shows that the collective action of agents emerged as opinion clusters, which in turn affects the local behaviors of the agents. A multi-agent model for simulating the spatio-temporal dynamics of a coupled human–landscape system demonstrated a cyclical influence, that is, agent interactions generate community and landscape changes and such macro changes create new opportunities or constraints for the agent-based processes, forming cross-scale feedback loops.[32] An agent-based vegetation model[7] demonstrated a cyclical process of micro to macro to micro influence of plant positions with regards to sunlight, shad, and nutrients, and a macro causal influence that was not produced by the micro states of the vegetation. In modeling culture with MASs, Morris et al.[38] state that:

> …the view of culture as a system promotes a focus on the emergence of culture from its tangible components, and how the relationships between these components openly affect the macro-level culture … and also how that macro-level culture in turn impacts and influences the behaviors of the micro-level components themselves. (p.4)

Not all complex systems carry a full cyclical micro–macro–micro feedback, however. Various case studies in global change ecology[31] show that microbial community activities can directly affect upper level processes (the meso level, if the term can be applied to ecology, but this depends on the perspective observations of the system, that is, the meso level here can be the macro level if the complex system being studied is bounded within that level, as in the relativity mentioned above by Gernstein[35]), which in turn influences the community, but will not affect the macro state of the system. For example, a shift in microbial community structure can result in a change in process rate (p.61). On the other hand, macro environments will definitely affect the micro level, for example, land use (tillage, forest clear-cutting, urban development) impacts the microbial community directly, all disrupting the soil and altering microbial abundance and functional types (p.66). Another example in ecology shows the macro to micro influence: Collembola in stumps are influenced by macrohabitat factors (stand age and extent geographic location).[30]

It is important to note the vast differences in the micro–macro link in society as compared to ecology, where cognitive capabilities of non-anthropomorphic organisms are at a minimum. Squazzoni[28] stated that the understanding of the micro–macro link requires grasping emergent properties, in two meanings – first-order emergent properties and second-order emergence. The first is a ''macro level property, i.e., a macro pattern, behavior, structure or dynamic, that is generated through decentralized and localized interaction among agents''. In the latter, a macro property is a second-order property only ''if it is cognitively recognized by agents that have yielded it and if, as a consequence, it can be intentionally supported, maintained, changed or contrasted by the same agents that yielded it''. In second-order emergence, the agents are aware of what they generate at the macro level. Whilst this article does not focus on the study of the micro–macro link, but rather, on a generic complexity modeling approach and the agent communications framework, the significance of the micro–macro integration in future simulation must not be disregarded and, therefore, the preceding paragraphs are included to cover the literatures. The simulation section (Section 4) does, however, relate the micro–macro link to the experiments.

Emergence, a macro-level property that is produced from the interaction of micro-level interactions, requires that the resolution of the model be of a sufficiently small scale. For this cause, the resolution of the model supported by the proposed model will need to be fine for it to reflect to a certain extent the reality that it attempts to model. For example, many fish schooling algorithms take account of the three basic rules – cohesion, alignment and avoidance. Fish tend to speed up and slow down autonomously based on the three simple rules. This model of schooling behavior has a fine resolution, but many other properties found in natural systems are unaccounted for. Microscopic organisms, such as unicellular organisms, frequently have strange dynamics such as sudden impulses and rhythms, while larger multicellular organisms have more coordination in their movements, and animals possessing a cerebral cortex make coordinated and intelligent movements. Living organisms consume energy, which affects their dynamics, ingesting food provides energy, and unused energy is

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

stored. Species richness can also be a matter of resolution – the more species there are in a given area, the higher the resolution of diversity. There is also the need for a means of identifying and tagging similar agents for purposes of mating, flocking, species formation, trophic network, social interaction and communication. These are some of the fine details that need to be addressed if we want to see fruition in models generating appropriate macro-level properties.

It is practical to approach the modeling of agents and their environments by the abstraction of their properties and principles into variables, states and methods associated with OO concepts. The OO modeling of Dynamics, Preference, Behavioral, Interaction, Communication and Social aspects (DPBICS) of a complex system facilitates the efficient reusability of similar code patterns for simulating agent processes via inheritance and polymorphism. Modelers can focus on DPBICS modeling if the foundation for OO is laid, which can be extended to include evolutionary traits[41] from concepts in artificial life[42] research.

2.3 Agent interaction and communication optimization

Past experience in modeling vegetation as agents[7,43] has shown that the computational complexity of the computing resource requirements increases proportionally to the population of agents and abiotic factors. Whilst traditional approaches in modeling the spatial distribution of species utilizes pixels as a patch size of n meters of space to describe large communities of organisms on a landscape, the ABM approach models individual agents and all their internal processes and external interactions. Experience tells us that even though the rules for an agent are simple, a higher resolution of modeling individual species using computer algorithms requires a large numbers of variables, algorithmic structures and computation. This is true even for a single entity. The availability of computing resources for simulation becomes an exponential challenge when biotic and abiotic interactions occur and entities reproduce. Resource bottleneck is a problem that requires solving before the simulation of large-scale complex systems can see fruition. Trophic networks, resource usage, social relationships and environmental factors are topics that are highly intertwined and are programmatically difficult even in small-scale ecosystems (limited number of population, resources and relationships). Agents sense the environment and access all global and local environmental factors, traverse the availability of food and resources at accessible proximity (there may be hundreds of thousands in the memory), and deal with a multitude of agent communication via tagging (friend, foe, food, etc.). Other challenges are the incorporation of ''timed'' species migration features into the system. How can we programmatically manage these interactions and communications in an efficient way? Unfortunately, a review of literature in the area yielded little evidence of such developments. Schulz and Reggia[44] developed a method for predicting nearest agent distances in artificial worlds. Other methods are developed for visualizing complex outdoor scenes in computer graphics,[45–47] but algorithms have not been formalized for improving the efficiency of large-scale interactions. Data structures for managing and categorizing large collections of data are available. The simplest, perhaps, is the array data structure. More advanced structures are hierarchical data structures, which are based on recursive decomposition.[48] Well-known hierarchical data structures[49] used in science and engineering for efficient representation and improving execution times are Binary Trees, Quadtrees and Octrees. Other efficient segmentation algorithms have been attempted[43] for sessile organisms. The depth of a trophic network is proportional to the computational complexity of the algorithm. Trophic networks are a form of communication in complex systems; the purpose of such communication is to allow the consumption of food in the lower chain. These issues have to be addressed before more ambitious models associated with vast landscapes can be resolved (e.g., Ch'ng et al.;[50] Gaffney et al.[51]). Later, the Quadtree data structure will be revisited.

The next section addresses three important strategies in modeling and simulating complex systems using computational methods: (1) resolution of the simulation – detail behavior and movement; (2) expandability of simulation – an adaptive Quadtree that supports efficient interaction; (3) optimization of algorithms – event-based simulation.

3. Strategies for modeling and simulating complex systems

It is predicted that there are ~8.7 million ($\pm$1.3 million SE) species on earth and in the ocean,[33] including 1.2 million species catalogued in a database, such a vast number of species means that generic modeling could be complicated. The diversity of life on Earth and the multiple layers of complex systems make it an intriguing area of study for the life sciences. Certain longer term processes, however, require looking at palaeoenvironmental data, while others draw conclusions from studies involving decades of monitoring a certain phenomena. However, observations can only yield so much in the little time that scientists persist on Earth. Simulations can be controlled, run in parallel but modified sessions, repeated to see how the modification of certain parameters changes the behavior of the simulation, and run for many simulated generations at compressed time. Such great control over synthesized life within computers has given researchers the means to eliminate the limitations of time. It has also allowed the omission of information noises by filtering parameters that are not required so that a better understanding of life can be

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

realized as a result of an unpolluted computer-generated environment.

The modeling presented in the subsequent sections is programming-language independent, although the code is in both C++ and Java with syntax and structuring variations between the two languages. Implementations in other OO languages are possible. Contents in the subsequent sections first deal with the functional hierarchy of the taxonomic classification, providing a method of generalizing the vast number of species differences into manageable classes that resolve both biological and algorithmic structures (Section 3.1). The section continues with a description of how the adaptive Quadtree can be used to manage vast numbers of trophic networks and agent–emitter interaction (Section 3.2). An event-based interaction for global and local climate is covered (Section 3.3). The next section deals with models of global climate and strategies for modeling local climate changes with emitter objects. The final section describes the model resolution of a single agent in detail, covering possible aspects of its behavior and how it relates to other agents in the network. Section 3.4 onwards focuses on the model resolution.

3.1 Classification resolution: functional object-oriented hierarchy

Two hundred and fifty years of taxonomic classifications have produced over a million species under the heading of life and its sub-headings (domain, kingdom, phylum, …, species). With ~7.4 million species predicted but unidentified, creating an OO inheritance under the present taxonomic structure would be irrational. A more rational means of structuring inheritance might be to model the classification in the way of generalization (Figure 1(a)) as done previously.[12] The advancement of modeling experience led to a more functional hierarchy proposed here in this article (Figure 1(b)). In Figure 1(a), organisms are generalized into vagile and sessile, with lower orders of life inheriting the properties and functions. The result is a complicated coding structure that is difficult to maintain and computationally inefficient. In Figure 2, developed in this research, considerations are given to merge functional properties between the biological agent, the rendering facilities and the interaction management. Using computer graphics as visualization, all Renderable objects (CRenderable – contains a render() method with properties such as visible, color, width and height) within the world inherit from CObject, so also utility classes for supporting the simulation (not shown here). The global environment (temperature, CO_2, etc.) classed as CEnvironment generates global conditions of the simulation model. CObject contains only the universal object ID and a toString() method. All objects that reside in the world that required interaction management are in the CWorldObject (see Figure 2). There need only be two different classes under the CWorldObject – CAgent and CEmitter, as all objects within the world belong to either class. CAgent represents all inherited agents within the world, and CEmitter represents a parent to all local environmental variations (see Section 3.3).

3.2 Adaptive Quadtree agent and emitter interaction management

The result of the functional OO hierarchy is the separation of the framework level with the coder's level. The OO framework-level codes are managed under the superclass CWorldObject, which is a class type that is stored in the TCollection class (in C++ a custom Template class that makes use of pointer memory for efficiency, in Java as a standard ArrayList collection class). The collection class needs to store vast numbers of agents and emitters. Ten million agents have been tested to work efficiently in the custom C++ collection class, subject to random-access memory (RAM) availability. OO polymorphism means that all subclass methods are called accordingly with virtual functions; this makes it a very efficient way of managing huge number of species during agent–agent agent–environment communications.

The adaptive Quadtree (Quadtrees are well-known data structures;[49] the implementation details are not within the scope of this paper) developed here is represented in Figure 3 as a structure with partially instantiated nodes. Instantiated nodes consume memory used by collection classes and associated variables. A landscape spanning thousands of kilometers may have kilometers of land that are uninhabitable and, as a consequence, may not have agents or emitters in that part of the terrain and, therefore, do not need to divide and instantiate new nodes. This saves memory and removes the need for interaction access for that particular node. The adaptive Quadtree implements a divide threshold and maximum layer capability, both of which can be set by users. The divide threshold determines the division of the nodes based on the number of agents and emitters in that node. The maximum layer decides the extent of the depth of the Quadtree. This gives custom control for a wide variety of simulation context (terrain type, agent categories, etc). Figure 3 shows agents (red) and emitters (green) in different nodes of the tree. Notice that the sequence of numbers is based on the sequence of node divisions. The sequence allows modelers to trace the early history of agents inhabiting the landscape.

There are two approaches for storing objects within the collection class. The first approach stores all agents and emitters in a global collection class and allows the Quadtree structure to manage only their indices. The second approach, used in this article, stores two collection classes (agents and emitters) within each node. The OO framework layer manages the segregation of agents and

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

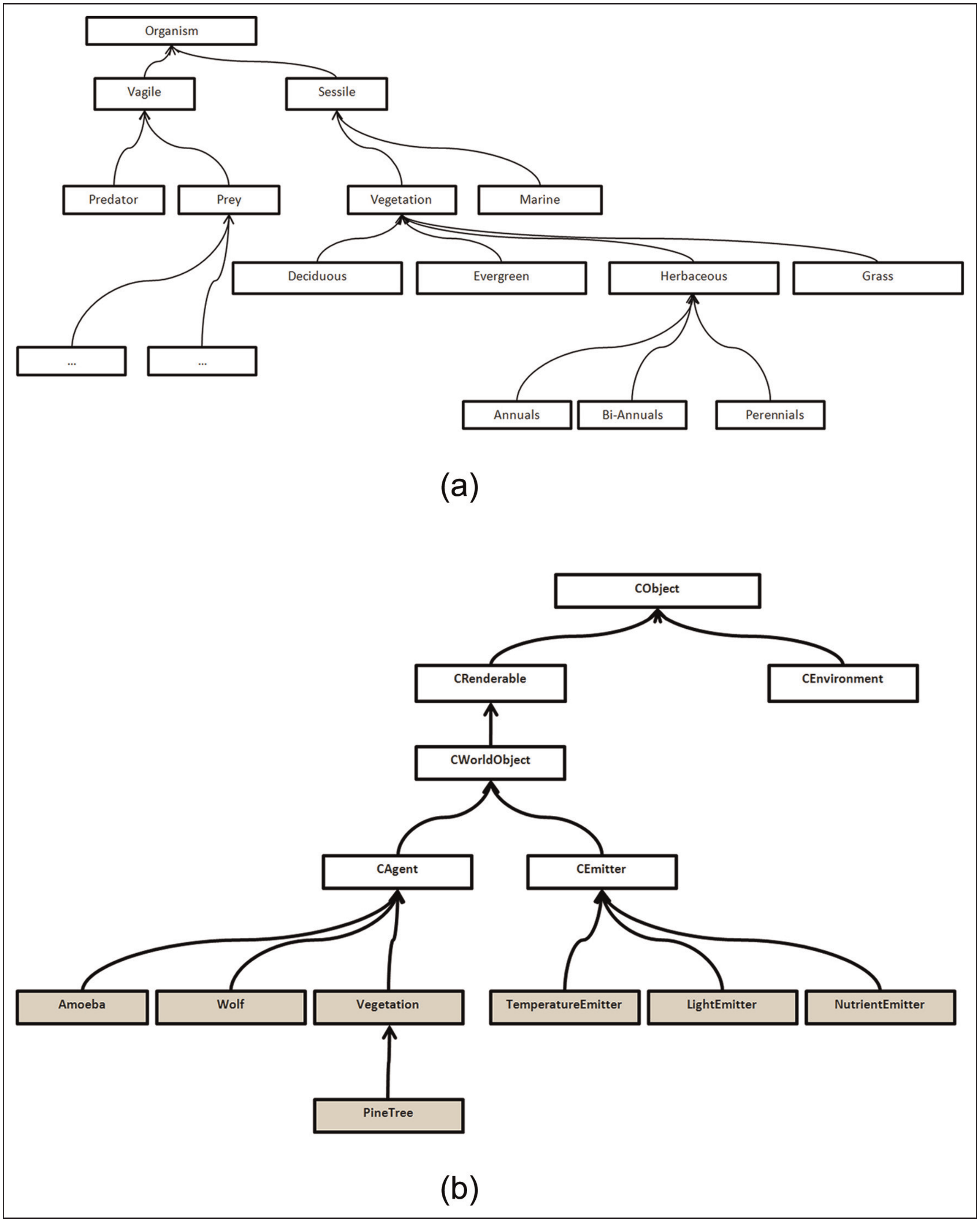

Figure 1. Object-oriented inheritance modeling of biological classification. (a) Compressed taxonomic classification. (b) Functional inheritance.

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

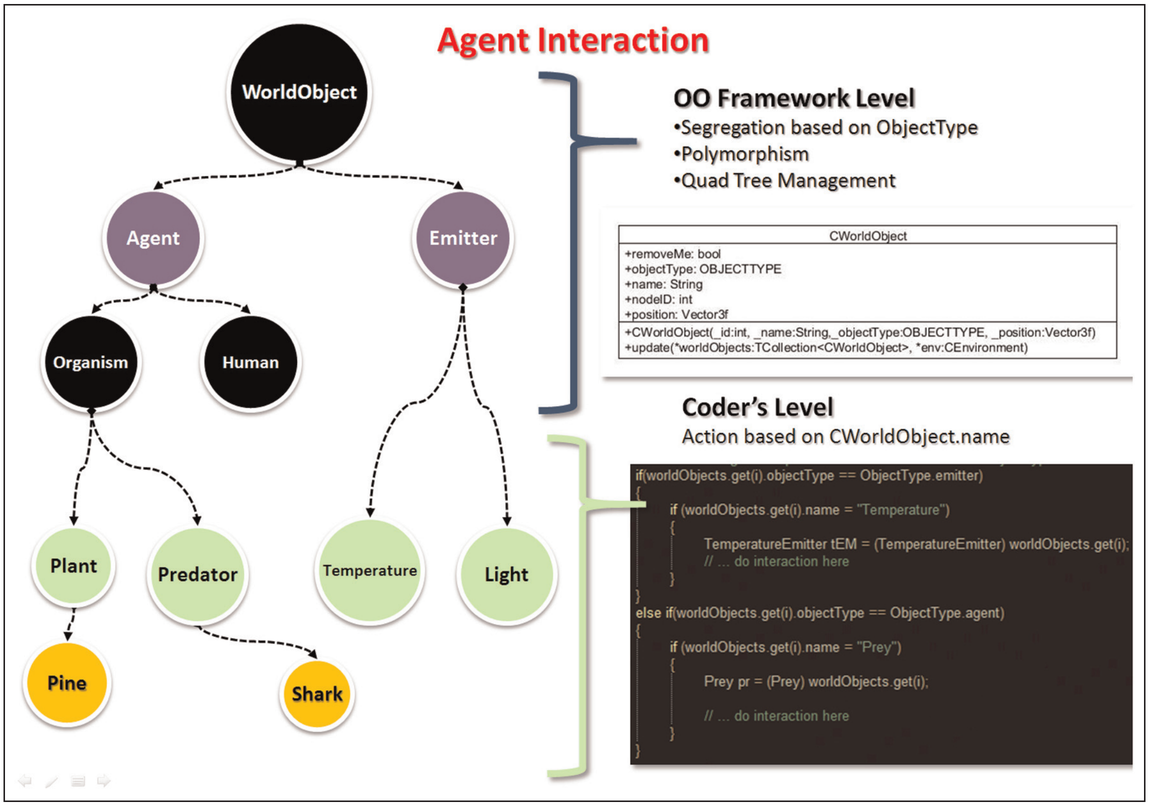

Figure 2. Object-oriented (OO) framework supporting Quadtree interaction. The framework has two levels – framework and coder's level. The framework level manages the Quadtree and how objects are segregated into object types. At the coder's level, modelers define the dynamics, preference, behavior and trophic network interaction of the agents.

emitters before assigning them to respective collection objects based on their types. It was found that assigning collection classes for individual nodes saves the Quadtree algorithm an additional step of having to access and transfer agent indices each time agents move to an adjacent node.

The adaptive Quadtree provides a function that queries and collects agents and emitters from nodes that are within a given radius (Figure 3). Collected agents are passed down to the objects managed by coders (Figure 2), which decide what custom type the object is, based on the name of the agent or emitter. Allowing coders to manage their own ''species'' opens up possibilities for unlimited types of species to be included in the model.

3.3 Event-based interaction in global and local climate

As stated earlier (Section 2.1), a complex environment creates complex behavior. Past models[7,11] implement only an average global climate; disregarding local climate means that species distribution will not be realistic. The approach taken in this research addresses the resolution of the model by the inclusion of both global and local climates. The Environment class, an extension of the CEnvironment superclass, manages the global climate. The local climate is both implicit and explicit and is managed by extension via the CEmitter class (e.g., TemperatureEmitter, NutrientEmitter, etc.). The signals generated by the global climate are consumed by agents via event subscription. The event subscription resolves issues with a large population of agents accessing the climate model. During runtime, as agents are being created, they subscribe to event models within the CEnvironment class. When any of the environmental factors change, the signal is sent to event subscriber agents. The sections below explain selected climate models used in the simulation in this article.

3.3.1 Global climate. The global climate is a macro state unaffected by micro-state properties. Figure 4 depicts a typical yearly cycle of environmental signals for temperature, sunlight, humidity and level of carbon dioxide with very minor variations throughout the months. Temperature reflects real world parameters and elevation, which is dependent on the coordinates of the plants. Sunlight, moisture and other factors are measured with the value in the range [0, 1] with full sun and water-logging/flooding at value 1.0. As the model is generic, more climatic factors

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

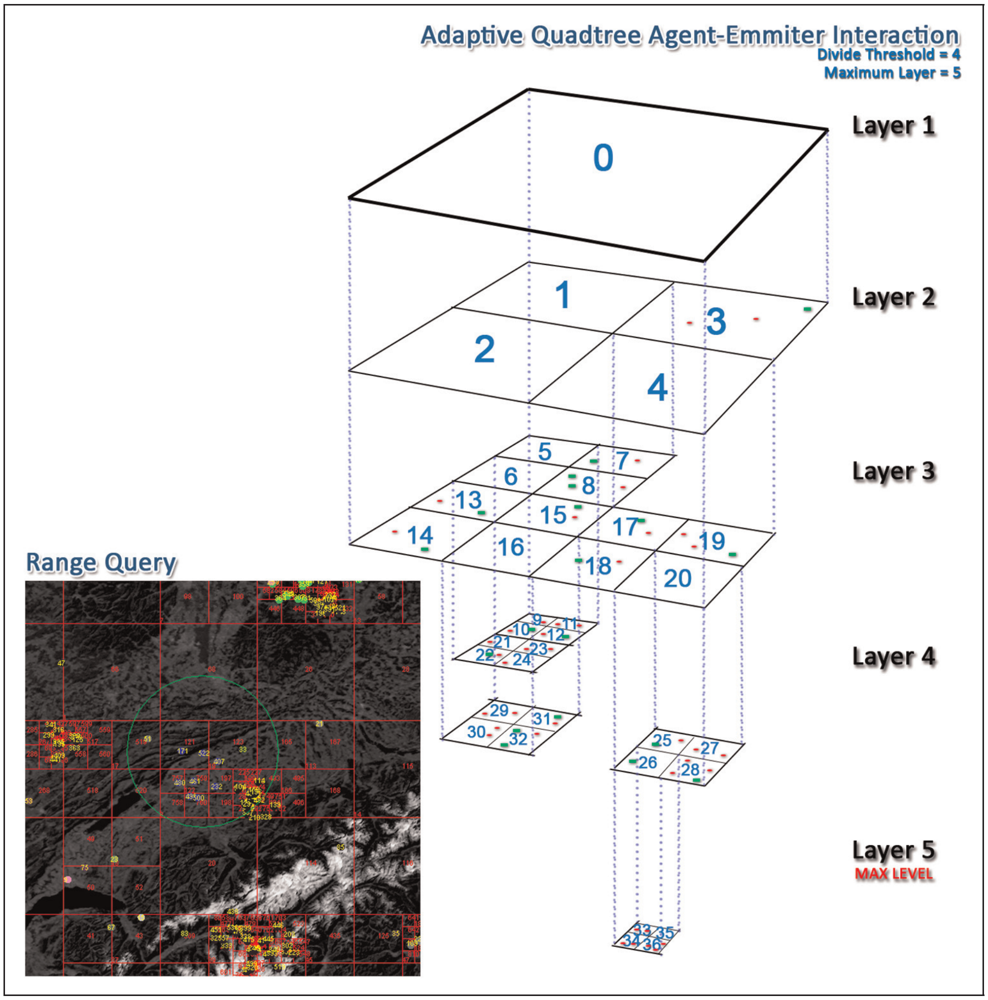

Figure 3. Adaptive Quadtree interaction supporting range query of agent–agent and agent–emitter interactions. The range query shows the range, represented as a circle, that an agent queries at a single time step in the simulation. (Color online only.)

can be included in the model by extending the Environment class.

The temperature model has a preset state defined by modelers, for example, $T_{pres} = \{0, 5, 10, 15, 20, 25, 25, 20, 15, 10, 5, 0\}$, with each element representing the average monthly temperature. A stochastic variable T_{var} determines the range of changes in the global temperature in every time cycle. Other environmental factors are defined in the same way. The equation below is an example of how a model Temperature works in the simulation:

$$T_{global} = \begin{cases} T_c + XT_{\text{var}} & \textit{if } X \leq g_c \\ T_c - XT_{\text{var}} & \textit{if } X > g_c \end{cases} \tag{0}$$

where $T_c \in T_{pres}$, X is a function from R to $[0, 1]$ and is a stochastic variable, T_{var} is the range of temperature change and g_c is a constant that defines the threshold when the temperature will change. The global temperature at a given time is in the range $-T_{\text{var}} + T_c \leq T_{global} \leq T_{\text{var}} + T_c$.

Temperatures in high altitudes can be modeled to decrease according to altitudinal limits, with a default of 0.6°C fall in temperature for each 100 m above sea level. The temperature–altitudinal ratio is

$$T_{eff} = \frac{-0.6E}{100} + T_{global} \tag{1}$$

where T_{eff} is the effective temperature, E is the current altitude where a plant is at and T_{global} is the seasonal global temperature defined in Equation (0).

Hydrology can also be modeled. Hydrology is measured in the range [0,1]. The distribution and increase of water is a continuous gradient to below the water surface so that marine vegetation and plants tolerant to water-logging can be simulated. The equation for hydrology in this particular virtual environment is defined with

$$W_{eff} = L\frac{1}{e^{\left(E - W_{surface}\right)g^{-1}}} \tag{2}$$

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

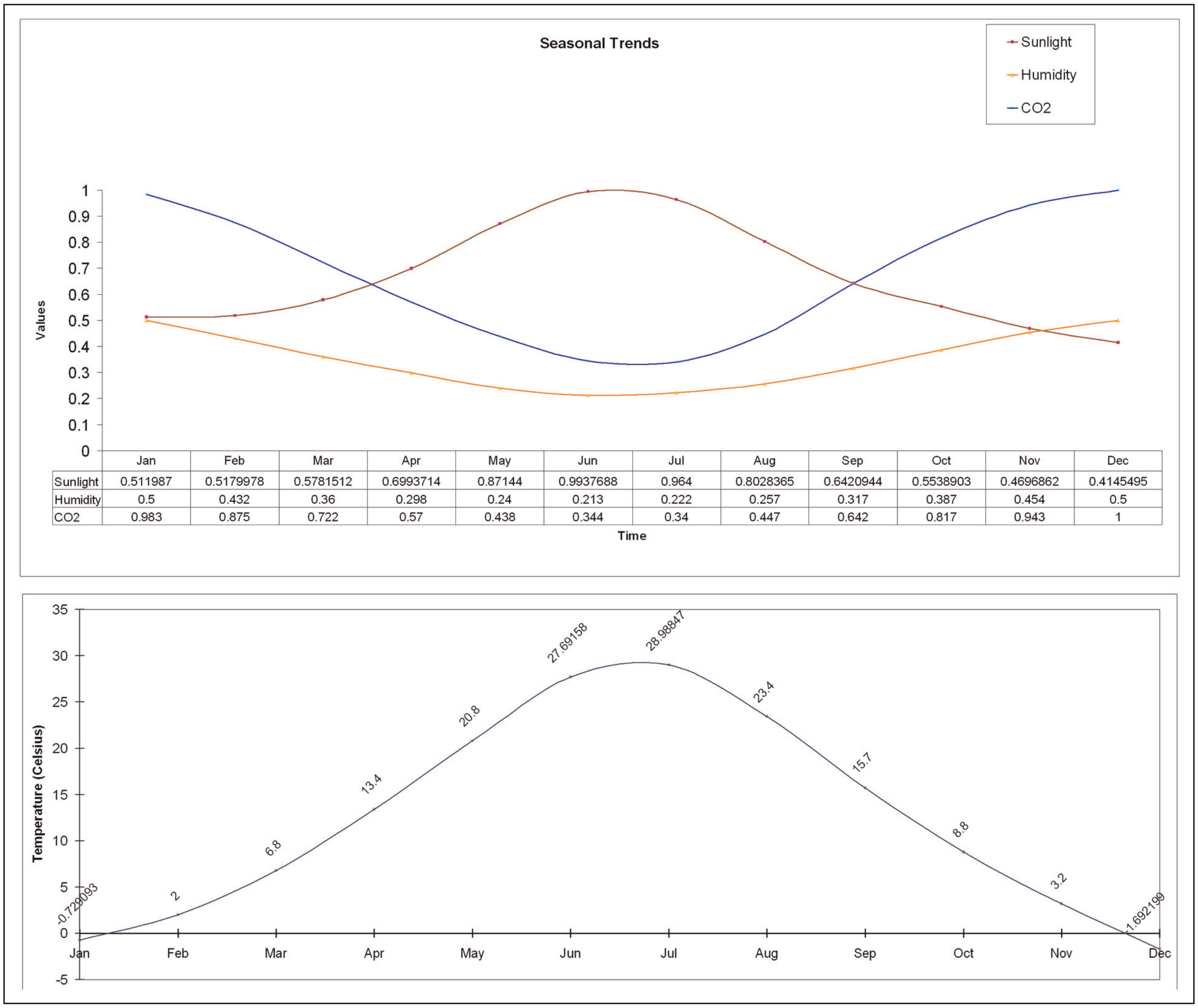

Figure 4. Graphs show in sequence the trends for sunlight, humidity, level of carbon dioxide and temperatures in the experiments.

where W_{eff} is the effective moisture level, L is the moisture level below the water surface (L = 0.5), E is the current altitude where a plant is at, $W_{surface}$ is the height of the water surface and g is the gradient.

The two environmental factors above are sufficient to demonstrate how other environmental effects can be simulated.

3.3.2 Climate resolution: local conditions. The local climate is a meso-level property that constrains or coordinates the behavior of agents. The local environment in which a plant draws its resources from is derived from three main sources. The first is implicit and is defined by the number of adjacent agents in the surroundings, which affects the effective sunlight, shade temperature, space and nutrient availability in a local area. An earlier paper on vegetation modeling describes the model. The second condition uses layered base maps (grayscale height maps) to define soil acidity, soil depth, ground textures or slope. The darker areas in soil acidity base map have lesser concentration. Lighter areas in the soil depth base map are shallower grounds and lighter areas in the ground texture have courser soils.

The third condition extends the CEmitter class and places emitter objects at strategic locations across the landscape. Agents at proximity with the emitters consume the local climate emitted by the local climate emitter. Emitter objects can have any number of parameters defined at the coder's level. The temperature emitter model of a single emitter below is an example:

$$\varepsilon_j = T_j\left(e^{d_j/g_j}\right)^{-1} \tag{3}$$

where ε_j is an emitter of temperature type at its core ($T_j = 100$ emits heat, $T_j = -100$ emits cold), d_j is the distance between the emitter and an agent, which dissipates

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

as the distance between the core and the agent increases, and g_j is the gradient. The total effective global and local temperature sensed by an agent is

$$T_{total} = T_{eff} + \sum_{j=0}^{n-1} \varepsilon_j \tag{4}$$

3.4 Model resolution: the individual agent

We have now come to the focus of the paper, which is to investigate ways in which an individual agent can be modeled. The format of the technical presentation of the model in this section follows that of the IBM technique in the Ecological Modeling community for easy readership.[52]

3.5 Purpose

This section addresses the model resolution of a complex system, at a fine micro-level resolution. It looks at the preference, movement dynamics, behaviors and trophic interactions of agents. The sections provide a fine resolution of models of agent behavior that sits within the framework defined earlier. The example model below will be used for various studies in Section 4.

3.6 Behavior resolution: state variables and scale

The state variable of the model defined below gives an example of the behavior resolution of the agents; it provides a basis for understanding the equations given soon after. It is a multi-level complex system comprised of a trophic network of individual, meta-population, community and environment. There are five types of agents represented here. The first two are vagile – consumers (prey, predator) – and the last three are sessile agents – producers that draw energy from sunlight and soil nutrients (herb, tree and poison). This web of interacting agents with a range of preferences, behavior and dynamics will demonstrate the robustness and resolution of the model.

Consumer agents are characterized by:

1. dynamics – speed (hunting, fleeing), eyesight, field-of-view (FOV);
2. physique – age, deterioration (affects the maximum age), energy, hunger threshold, flesh index;
3. behavior – impulse, safe distance (security boundary), feeding distance;
4. reproduction – number of progenies, reproduction distance, probability and sexual maturity;
5. fitness and adaptation – adaptability to sunlight, temperature, seismic vibrations and humidity.

Vegetation properties are characterized by energy, resource index, seed count, reproduction age, dispersal distance and ecology – adaptability to sunlight, temperature, soil *pH*, availability of space and humidity.

The populations are characterized by size (the number of individuals in a given species) and the community interacts within their trophic network. Table 1 shows an overview of the processes, parameters and individual genotype of the model. The symbols used represent variables in the equations in the Section 3.11. They are presented here as examples for demonstrating the resolution of the models in section 4. All parameters stored as Extensible Markup Language (XML) files can be tweaked in different simulations.

3.7 Process overview and scheduling

Time is discrete in the model and the aging of the agents proceeds in assigned milliseconds. Within each second certain processes (simple rules) occur – micro-state changes, growth, interaction, adaptation, feeding/fleeing, reproduction and inheritance, and senescence. The processes for each agent, which are autonomous, are specified in the pseudocode below.

For Producers
Sense the environment (Temperature, Sunlight, Humidity, pH level, Space)
Grow (aging) and die of senescence (Maximum age)
Compete with nearby plants for space
Reproduce when sexual maturity is reached
Die when the fitness is at a threshold
For Consumer (Predator)
Sense the environmental (Temperature, Sunlight, Humidity, earthquake)
Change states
Grow (aging) and die of senescence
Avoid Tree and Poison
Reproduce when sexual maturity is reached
Die when the fitness is depleted
For Consumer (Prey)
Sense the environmental (Temperature, Sunlight, Humidity, earthquake)
Change states
Grow (aging) and die of senescence
Avoid Predator
Change color when toxic plant eaten
Reproduce when sexual maturity is reached
Die when the fitness is depleted

3.8 Design concepts

Emergence: emergence occurs at the population level where interactions occur between the same species of agents. At the community level, certain emergent phenomena are observable due to the interactions between predator, prey, vegetation (food and protection) and abiotic factors.

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

Table 1. Overview of processes, parameters and individual genotypes of the model.

Global parameters	Symbols and value(s)
Aging of agents (per seconds)	1000 ms
Simulation time steps (dynamics)	Δt = 0.012 s
Rendering frames per seconds (FPS)	12
Abiotic parameters	Symbols & value(s)
Temperature	25°C
Sunlight	0.6
pH	8
Humidity	0.5
Seismic vibrations	0 or 1
General agent parameters	
Tagging (agents decide which action to take based tags)	Food [J_1, J_2, …, J_n] Predator [P_1, P_2, …, P_n] Spec [S_1, S_2, …, S_n]Etc.
Consumer (predator) parameters	Symbols & value(s)
Age at start of simulation (seconds)	1
Maximum age	A^{max} = 60
Deterioration time (a threshold reached where the creature deteriorates)	10
Speed (pixels)	σ = 2
Thrust	ε = 2
Hunting thrust	ε = 3
Fleeing thrust	ε = 4
Thrust limit	τ^{lim} = 0.5
Friction	ϕ = 0.9
Rotation angle	θ = 10°
Eyesight	D = 100
Field of view	v = 90°
Energy	λ = 1.0
Energy death threshold	$\lambda_j^{death} = 0.0$
Energy loss (when moving)	$\kappa = -0.001$
Energy rest threshold (at a point when agent needs rest)	υ = 0.2
Energy used in predation	$\kappa = -0.6$
Fitness death threshold	$f_j^{death} = 0.0002$
Hunger threshold (depletion of energy results in hunger)	0.5
Flesh index	$\xi = 1.0$
Impulse range (pixels)	ι = 200
Impulse	I = 20
Safe distance (from predation)	80
Feeding distance (prey captured at this distance)	10
Progenies	n = 1
Reproduction threshold	$R_j^{thr} = 3.5$
Reproduction fitness	$R_j^{fitness} = 0.3$
Reproduction age	$R_j^{age} = 1$
Sunlight (L = lower range, P = preferred, U = upper range)	L:0.2, P:0.6, U:0.8
Temperature (°C) (L = lower range, P = preferred, U = upper range)	L:–4.0, P:25, U:35
Humidity (L = lower range, P = preferred, U = upper range)	L:0.3, P:0.5, U:0.8
Seismic vibration (earthquake)	1
Consumer (prey) parameters	Symbols & value(s)
Age at start of simulation (seconds)	1
Initial size of creature at birth	ς^0 = 0.5
Rate of growth	g = 5.0
Maximum age	A^{max} = 100
Deterioration time (a threshold reached where the creature deteriorates)	10
Speed (pixels)	σ = 2
Thrust	ε = 3
Hunting thrust	ε = 3
Fleeing thrust	ε = 8
Thrust limit	τ^{lim} = 0.5
Friction	ϕ = 0.9
Rotation angle	θ = 10°
Eyesight	d = 250

(continued)

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

Table 1. Continued

Global parameters	Symbols and value(s)
Field of view	v = 90°
Energy	λ = 1.0
Energy death threshold	$\lambda_j^{death} = 0.0$
Energy loss (when moving)	$\kappa = -0.0005$
Energy rest threshold (at a point when agent needs rest)	υ = 0.0
Energy used in predation	κ = 0.5
Fitness death threshold	$f_j^{death} = 0.0002$
Hunger threshold (depletion of energy results in hunger)	0.8
Flesh index	$\xi = 1.0$
Impulse range (pixels)	ι = 200
Impulse	I = 5
Safe distance (from predation)	*50*
Feeding distance (prey captured at this distance)	3
Progenies	$n = 5$
Next reproduction age (reproduced only after *n* seconds)	15
Sunlight (L = lower range, P = preferred, U = upper range)	L:0.1, P:0.6, U:0.8
Temperature (°C) (L = lower range, P = preferred, U = upper range)	L:–4.0, P:25, U:35
Humidity (L = lower range, P = preferred, U = upper range)	L:0.3, P:0.5, U:0.8
Seismic vibration (earthquake)	1
Producers (herb) parameters	**Symbols & value(s)**
Age at start of simulation (seconds)	1
Maximum age	A = 70
Energy	ρ = 1.0
Resource index	K = 1.0
Seed count	s = 5
Reproduction age	10
Dispersal distance (pixels)	D = 130
Sunlight (L = lower range, P = preferred, U = upper range)	L:0.1, P:0.5, U:0.8
Temperature (°C) (L = lower range, P = preferred, U = upper range)	L:18.0, P:26, U:38
Soil (P = preferred, U = upper range)	P:0.5, U:0.8
pH (L = lower range, P = preferred, U = upper range)	L:5.0, P:8.0, U:10.0
Space (P = preferred, U = upper range)	P:0.6, U:1.0
Humidity (L = lower range, P = preferred, U = upper range)	L:0.3, P:0.5, U:0.7
Producers (tree) parameters	**Symbols & value(s)**
Age at start of simulation (seconds)	1
Maximum age	A = 150
Energy	ρ = 1.0
Resource index	K = 1.0
Seed count	s = 5
Reproduction age	25
Dispersal distance (pixels)	D = 250
Sunlight (L = lower range, P = preferred, U = upper range)	L:0.1, P:0.8, U:1.0
Temperature (°C) (L = lower range, P = preferred, U = upper range)	L:15.0, P:30, U:45
Soil (P = preferred, U = upper range)	P:0.4, U:0.6
pH (L = lower range, P = preferred, U = upper range)	L:1.0, P:7.0, U:10.0
Space (P = preferred, U = upper range)	P:0.28, U:0.5
Humidity (L = lower range, P = preferred, U = upper range)	L:0.3, P:0.5, U:0.7

Adaptation: adaptation in the model refers not to the general adaptation in evolutionary time-scale, but to the tolerance of biotic and abiotic factors that have already been developed.

Fitness: the fitness of each organism is measured by the Adaptability Measure (AM).[10] The agents are affected by three abiotic factors: temperature, sunlight and humidity; the plants are affected by six factors: temperature, sunlight, soil, humidity, pH level and space. Soil conditions are unchanged and are based on height fields (certain regions are more habitable). The availability of space for the plants depends on the number of plants growing within that space. Fitness-seeking is not modeled. The fitness measure is provided in later sections and is a product of the AM.

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

Prediction: the estimation of future consequences of agent decisions is not modeled.

Sensing: agents are aware of their own age-related mechanisms (reproduction, senescence), other agents that have direct impact on their survivability and the abiotic factors.

Interaction: agents sense all agents within the landscape but tagging by name in the CWorldObject decides which agents to interact with.

Stochasticity: stochasticity is implemented in the distribution of seeds (angle and bounded distance) and the impulses of the movement thrusts of the vagile agents.

Collectives: individuals are grouped into collectives or social groups via tagging. Explicitly grouped agents flock, but environmental niches implicitly cluster agents together.

Observations: observations of the model are provided later in this article.

3.9 Initialization

At the start of simulation, all agents are randomly distributed in the landscape using the default parameters listed in Table 1. Abiotic factors, such as global and local climate, are initialized and read from the preset conditions.

3.10 Input

Abiotic parameters are directly read from the environment and emitter classes. This corresponds to the simulation cycle of the agents, ensuring a continuous flow of abiotic information to the biotic components.

3.11 Submodels

This section explains in detail all submodels and parameterization representing the processes listed in Table 1. The section covers the mathematical structure of agent preferences, behavior and dynamics.

3.11.1 Producer behavior. This section describes the behavior of vegetation.

Sensing the environment

The vegetation agents sense the environment via receptors. The abiotic factors that the agents take into account are temperature, sunlight, humidity and soil pH. Biotic factors include the competition for space and the risk of being consumed by prey. Each abiotic factor is measured by the AM[10] and contributes to the fitness measure.

Fitness measure and death

The AM measures four abiotic factors and one biotic factor to generate the fitness for individual agents i at time t (Equation (5)) in each simulation cycle:

$$f_i^t = \varphi_i C_i^t T_i^t S_i^t \eta_i^t \tag{5}$$

where the output of AM for each fitness related to biotic or abiotic interactions are computed: $\varphi_i$is constant at runtime is the interaction fitness of the pH level of the soil for plant i, C_i^t is the only biotic local interaction fitness of the current condition c_i^t (Equation (2)) at time t, T_i^t is the interaction fitness of the plant related to the temperature, S_i^t is the fitness affected by the sunlight and η_i^t is the fitness of the plant in the current humidity. The interaction of the factors is a logical way for deciding the fitness of the plant. The variable Resource Index ρ, defined as the storage of energy is decremented κ unit (Table 1) in the condition in Equation (6), is defined partly with the Iverson bracket. Death occurs when $\rho_i^{t+\Delta t} \leq 0.0$:

$$\rho_i^{t+\Delta t} = \rho_i^t - \left(\kappa_i \left[f_i^t \leq f_i^{death}\right]\right) \tag{6}$$

Competition

The collective occupation of the space used by the competing plants contributes to the accumulated space c_i^t at time t for the plant i in Equation (7). Competition for space is defined as an interaction. A plant interacts with its neighbor in the condition

$$c_i^t = \sum_{i=1}^{n} P_i \left[\sqrt{(O_x^t - u_x^t)^2 + (O_y^t - u_y^t)^2} - \left(O_{size}^t + u_{size}^t\right) < 0\right]$$
$$\left[O_{size}^t \geq u_{size}^t\right]\left[O_{age}^t \geq u_{age}^t\right],$$
$$\text{for } 0 \leq c_i^t \leq 1 \tag{7}$$

where n is the number of competing plants and P_i is the effective space used by a single plant. $P_i = 0.05$ if the undergrowth species (herb or poison) competes against its own species, $P_i = 0.12$ if the undergrowth compete against a tree and $P_i = 0.2$ among tree competition. The differences in P_i adjusts the space so that the canopies are not too crowded together. $O_{x,y}^t$ is the position of the competitor and $u_{x,y}^t$ is the position of the source plant at time t. O_{size}^t and u_{size}^t, O_{age}^t and u_{age}^t are, respectively, the diameter and the age of the two competing plants.

Growth and reproduction

Growth and reproduction uses time as a measurement. Growth is defined by aging – the plant ages in milliseconds and senescence occurs when the maximum age is reached. Reproduction for Trees and Herbs depends on the parameter Reproduction Age and the number of seeds s. The size of the plant has the following ratio:

$$\varsigma_j^{t+\Delta t} = \varsigma_j^{\max}\left(1 + e^{g\left(1 - \frac{4t}{A_j}\right)}\right)^{-1} \tag{8}$$

where g is a constant (g = 5.0) is the *Rate of Growth* to reach full size, it is the growth spurt, A_j is the *Maximum Age* of the agent, and $\varsigma_j^{\max}$ is the maximum size of plant j.

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

The growth–seed ratio is modeled using the equation below:

$$G_j^{t+\Delta t} = s_j \varsigma_j^t \tag{9}$$

where $G_j^{t+\Delta t}$ is the total number of seeds for agent j at time t and s_j is the maximum seed count of the agent. Reproduction is timed: the first occurrence of reproduction is when the agent matures in age. The parameters for the variables below are listed in Table 1.

Reproduction distance and direction are stochastic:

$$x_s = x_i + D_i X \cos\beta \tag{10}$$

$$y_s = y_i + D_i X \sin\beta \tag{11}$$

where x_s and y_s are, respectively, the x and y position of the seed, x_i and y_i are, respectively, the x and y position of the parent plant, X is a function from R to $[0, 1]$ and is a stochastic variable. D_i is the maximum dispersal distance of the parent plant (see Table 1), and finally β is a random angle in the interval $[0, 360]$.

3.11.2 Consumer behavior.

Sensing the environment

The agents sense the environment via receptors. The abiotic factors that the agents take into account are temperature, sunlight and humidity. Each abiotic factor is measured by the AM[16] and contributes to the fitness measure.

Fitness measure

The AM measures each environmental factor, which is combined with five factors to generate the fitness f_j^t for each individual agent j (Equation (12)) in each simulation cycle:

$$f_j^t = T_j^t S_j^t \eta_j^t \tag{12}$$

where the output of AM for each fitness related to abiotic interactions is computed: T_j^t is the interaction fitness of agent j at time t in the current temperature, S_j^t refers to the interaction fitness with sunlight and η_j^t is the fitness in the current humidity.

Growth, reproduction and death

Growth and reproduction uses time as a measurement. Growth is defined by aging – the agent ages in milliseconds and senescence occurs when the maximum age is reached. Graphically, the size of the prey agents has the following ratio, while predator size does not change. The formula used for growth is according to Equation (8).

Reproduction depends on the parameter *Mature Age Ratio, Reproduction Age* and the number of *Progenies n*. The progenies appear from below the agents when they are reproduced. The parameters for the variables below are listed in Table 1:

$$Q_j^t = \left[\lambda_j^t > R_j^{thr}\right]\left[f_j^t > R_j^{fitness}\right]\left[A_j^t > R_j^{age}\right], \lambda_j^t = \sum_{k=1}^{n} \xi_k \tag{13}$$

where Q_j^t is a Boolean variable deciding the reproduction of consumers, which depends on the flesh index ξ_k consumed of the captured prey k surpassing the reproduction threshold R_j^{thr}, fitness $R_j^{fitness}$ and age R_j^{age} of agent at time t.

Death occurs when the conditions below are fulfilled:

$$D_j^t = \left[\lambda_j^t < \lambda_j^{death}\right]\left[f_j^t < f_j^{death}\right]\left[A_j^t \geq A_j^{\max}\right] \tag{14}$$

where D_j^t is a Boolean variable deciding agent death, which depends on the fitness and energy decreasing to below f_j^{death} and λ_j^{death}, and the age surpassing $A_j^{\max}$.

3.11.3 Movement dynamics of consumers.

Movement when targeting another agent

During one simulation time step, the steering behavior of the agent position updates with the speed, thrust (to propel the body forward) and heading:

$$x_j^{t+\Delta t} = x_j^t + \sigma_j \tau_j^t \cos(\omega_j^t) \tag{15}$$

$$y_j^{t+\Delta t} = y_j^t + \sigma_j \tau_j^t \sin(\omega_j^t) \tag{16}$$

where $\left[x_j^{t+\Delta t} \quad y_j^{t+\Delta t}\right]$ is the vector of agent j with speed σ_j (length of the vector) and *Thrust* τ_j^t (changes only during the roaming state, see Equation (27)) at time $t+\Delta t$. The heading ω_j^t (orientation of the vector) depends on the position of the target agent j:

$$\omega_j^t = \begin{cases} +\theta_j & \text{if } u_{jk} > 1 \\ -\theta_j & \text{if } u_{jk} < -1 \end{cases} \tag{17}$$

where θ_j is the agent heading. Avoidance behavior is defined below by simply swapping the sign associated with the heading:

$$\omega_j^t = \begin{cases} -\theta_j & \text{if } u_{jk} > 1 \\ +\theta_j & \text{if } u_{jk} < -1 \end{cases} \tag{18}$$

u_{jk}^t and v_{jk}^t is the visibility (*Eyesight*) from agent j to target agent k with the distance between the agents in the x and y component x_{jk}^t and y_{jk}^t at time t:

$$u_{jk}^t = x_{jk}^t \cos(-\omega_j^t) + y_{jk}^t \sin(-\omega_j^t) \tag{19}$$

$$v_{jk}^t = -x_{jk}^t \sin(-\omega_j^t) + y_{jk}^t \cos(-\omega_j^t) \tag{20}$$

where

$$x_{jk}^t = x_k^t - x_j^t \tag{21}$$

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

$$y_{jk}^t = y_k^t - y_j^t \tag{22}$$

The visibility of the agent from agent j to k and its position x and y at time t are

$$\left\| d_{jk}^t \right\| = \sqrt{(x_j^t - x_k^t)^2 + (y_j^t - y_k^t)^2} \tag{23}$$

The target agents are visible to the hunting agents at 180° FOV:

$$\vartheta_j = \left[v_{jk} > 0 \right] \tag{24}$$

where ϑ_j is the decision to hunt (move to) the target.

Movement during the ROAM state

When in the roaming state, both predator and prey use their impulses to navigate. The impulse to turn either right or left and the thrust to move forward in short bursts depends on Equations (25) and (26):

$$\omega_j^t = \begin{cases} +\theta_j & \text{if } X\iota_j < \mathrm{I}_j \\ -\theta_j & \text{if } X\iota_j > \mathrm{I}_j \end{cases} \tag{25}$$

$$\tau_j = \begin{cases} \varepsilon_j & \text{if } X\iota_j < \mathrm{I}_j \\ 0 & \text{otherwise} \end{cases} \tag{26}$$

where ω_j is the heading of the agent j with steering angle θ_j at time t, the thrust τ_j is the decision to propel the body forward with force ε_j, X is a function from R to $[0, 1]$ and is a stochastic variable, ι_j is the *Impulse Range* and I_j is the *Impulse* of the agent (in Table 1). During the roaming state a friction is applied to the thrust for the simulation time step $t + \Delta t$:

$$\tau_j^{t+\Delta t} = \begin{cases} \tau_j^t \phi_j & \text{if } \tau_j^t \phi_j < \tau_j^{\lim} \\ \tau_j^{\lim} & \text{otherwise} \end{cases} \tag{27}$$

where τ_j^t is the *Thrust* in the *Thrust Limit* $\tau_j^{\lim}$ and ϕ_j is the *Friction* applied to the force.

Energy gain and loss

Energy is gained via predation and food intake and energy loss occur during the movements. The use of energy depends on the efficiency of each species with the gain and loss:

$$\lambda_j^{t+\Delta t} = \lambda_j^t + \kappa \tag{28}$$

where $\lambda_j^{t+\Delta t}$ is the energy of the agent j at time $t + \Delta t$ and κ, defined in Table 2, is the amount of energy gained or lost. When a predator acquires prey as food, the *Flesh Index* (Table 1) ξ is added to the energy of the predator. The preys acquire their energy from the *Resource Index* of the plants. The agent rests (stops activities) in the condition $\lambda_j^t < \upsilon_j$, where υ_j is the *Energy Rest Threshold* (Table 1).

Collective behavior

Agents can act collectively, based on a series of summed positional vectors. Continuing from the movement dynamics in Equations (15) and (16), the collective dynamics of an agent based on other agents' dynamics in proximity is

$$x_j^{t+\Delta t} = x_j^t + \sum_{l=0}^{z-1} q_l \tag{29}$$

$$y_j^{t+\Delta t} = y_j^t + \sum_{l=0}^{z-1} w_l \tag{30}$$

where x_j^t and y_j^t are the vectors of agent j (Equation (15) and (16)), and q_l and w_l are a collection of average positional and directional vectors of agents k within the range and FOV (Equations (19)–(24)) of agent j. A variant of the basic flocking behavior used in Section 4, for example, would have $z = 3$. The z count can be increased to include other vectors such as target or wind vectors that affect or manoeuvre the collective agents' dynamics. The cohesion (l = 0) with coefficient C_{ratio}^{-1} for the ''cohesiveness'' of the flock is in Equation (32). Separation (l = 1) in Equations (33) and (34) occurs only when agents k are at proximity (usually <100) with coefficient S_{ratio}^{-1}. The alignment (l = 2) vector in Equations (35) and (36) calculates the average direction of agents at proximity.

$$q_0 = C_{ratio}^{-1} \left(n^{-1} \sum_{k=1}^{n} x_k \right) - x_j \tag{31}$$

$$w_0 = C_{ratio}^{-1} \left(n^{-1} \sum_{k=1}^{n} y_k \right) - y_j \tag{32}$$

Table 2. Energy gain and loss for both predator and prey.

States	-κ Loss (prey)	-κ Loss (predator)	$+\kappa$ Gain (prey)	$+\kappa$ Gain (predator)
ROAM	0.0001	0.001	–	–
HUNT	0.0001	0.001	–	–
FLEE	0.001	0.001	–	–
CHASE	–	0.001	–	–
SEARCH	0.0001	–	–	–
HIDE	0.0001	0.001	–	–
REST	–	–	0.1	0.01

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

$$q_1 = -S_{ratio}^{-1} \sum_{k=1}^{n} x_k - x_j \tag{33}$$

$$w_1 = -S_{ratio}^{-1} \sum_{k=1}^{n} y_k - y_j \tag{34}$$

$$q_2 = n^{-1} \sum_{k=1}^{n} \cos(\theta_k) \tag{35}$$

$$w_2 = n^{-1} \sum_{k=1}^{n} \sin(\theta_k) \tag{36}$$

4. Model simulation

This section tests separate aspects of the model resolution within the OO framework. As it is difficult to demonstrate the full potentials of the framework within an article, specially selected scenarios are used for demonstration. It is also difficult to portray movement dynamics, behavior and trophic networks within an article. The task of interpreting the results in terms of ecology is beyond the scope of this paper. The scenarios are presented as facts for demonstrating the model resolution and potentials of the framework in its supporting fine-resolution models and massive interaction and communications. The micro–macro link of the simulations is also presented at the end of every section.

4.1 Quadtree interaction optimization

Here, the adaptive Quadtree supporting a model is tested. The goal of the simulation is to see what the population of vegetation the framework will support at the 3.*n* seconds per cycle of time it takes an agent to do all its actions in DPBICS (see Section 2.2). The simulation models 255 species of vegetation growth with unlimited carrying capacity in the landscape. The results are shown in Figure 5. In the simulation where each agent does a range query of 100 pixels, at a cycle of 3.1 seconds, the framework supports 7376 agents. In a simulation where every agent has a 30 pixel radius of range query, at 3.25 seconds, the framework supports 10,399 agents. In another simulation not shown in the graphs (logarithmic scale), without using the OO framework's adaptive Quadtree, the simulation runs at 3.2 seconds per cycle for 5120 agents. The second simulation with a smaller radius of range query supports a faster per-cycle simulation.

4.2 Global and local climate

All simulations in Section 4 support global climate. This section focuses on the communications between agents and emitters in a simple simulation. Environment emitters, a preset meso-level property that affects the macro condition of the environment, bring a finer resolution to the local climate of a landscape. In the simulation shown in Figure 6, two environmental emitters are positioned in the

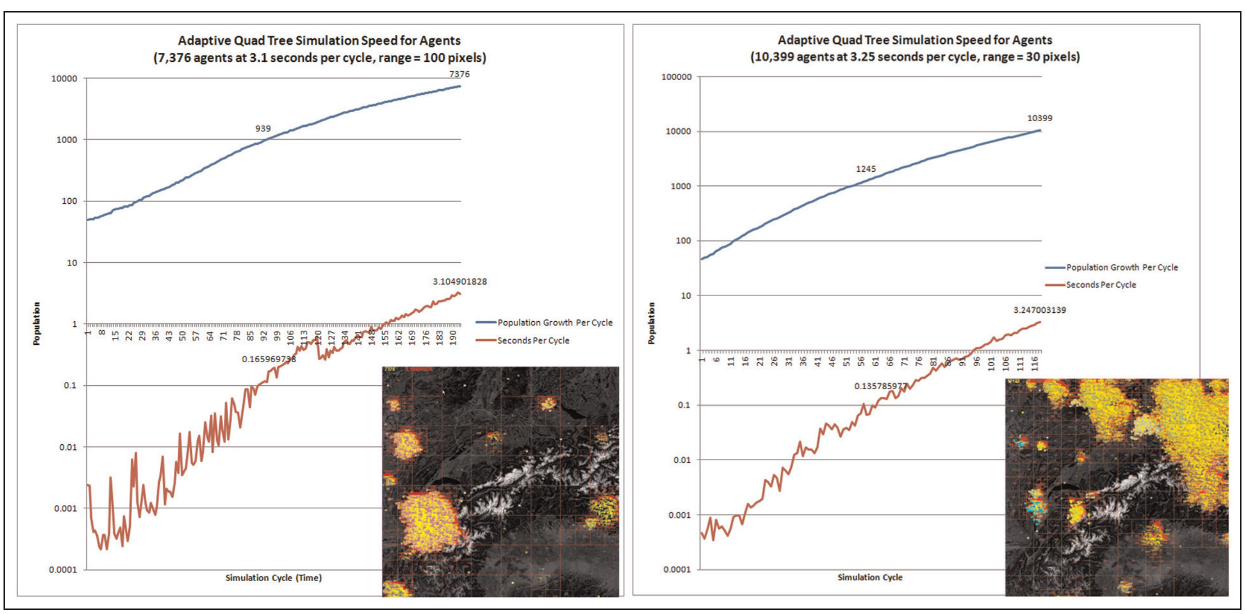

Figure 5. Quadtree results showing two log-plots of simulation within the object-oriented (OO) framework. In the simulation where each agent does a range query of 100 pixels, at a cycle of 3.1 seconds, the framework supports 7376 agents. In a simulation where the agent has a 30-pixel radius of range query, the framework supports 10,399 agents.

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

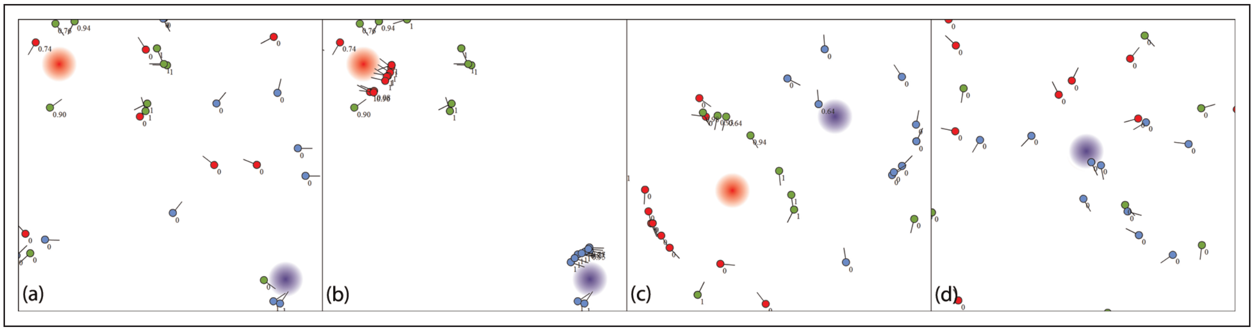

Figure 6. Introducing emitters (local climate) into simulations produces interesting phenomenon. (a) Agents roam the landscape looking for niches. (b) Agents settling down in their respective niches. (c) The interplay of temperatures between the two emitter cores causes agents to move in an emergent pattern. (d) The cores cancel each other out and agents roam freely, unable to find their niches. (Color online only.)

simulation. The red emitter emits heat and the blue emitter emits cold. Agents in red, blue and green are attracted to temperatures in their respective colors based on an extension[53] of the formula in Section 3.11.3. The green agents are attracted to intermediate temperatures between heat and cold. In Figure 6(a), the agents travel through the landscape, finding their niches. In Figure 6(b), the agents are stationed at their niches. In Figure 6(c), as the emitters are brought closer together, the agents are no longer stationary, rather they move in circles surrounding the source of their niche. In Figure 6(d), the two emitters are brought together and they cancel each other out. As a result, the agents roam freely, unable to find niches to settle down in. With regards to the micro–macro link, the simulation shows that the emitters have direct causal influence on the micro states of the agents; shifting the position of the local climate changes the macro condition of the landscape, which alters the movement patterns of the agents.

In a larger population simulation with a local climate using emitters, different species of agents are seen to avoid a wall of emitters, with occasional crossing of boundaries into the inner and outer environments. Heat emitters are positioned in a circle within the terrain, trapping the 120 agents inside and outside of the ''wall'' of emitters. After 1621 simulation cycles, the movement trails of agents can be seen to concentrate on the outer part of the terrain, and create a vortex of circling agents at the center (Figure 7). The interaction of agents, namely avoidance and predation, adds to the jittering movement patterns. In this simulation, the emitters completely transformed the macro condition of the environment. The wall of emitters acts as a protection for prey that are adaptable to the heat whilst repelling predators. The simulation demonstrated a macro–micro influence, which alters the collective macro patterns of predator–prey and emitter interaction in the environment.

4.3 Simple collective behavior

The simulation in Figure 8 shows the common flocking behavior from the local information communication of agents (avoidance, cohesion, separation). The separation variable is set at 80, 20 and 0, respectively, for Figures 8(a)–(c), using the collective behavior in Equations (29)–(36). At 1, the agents show erratic behavior and jittering as they compensate for the separation space between the agents. Here, the individual agents are micro-level entities and their simple micro-level mechanisms of interaction resulted in a macro-level behavior. The state of the macro-level behavior affects the movements of the micro-level entities, resulting in a cyclical influence of micro–macro to micro effects.

4.4 Simple trophic network

The simulation in Figure 9 shows a simple trophic network of three species consisting of two consumers and a producer – predators, prey and phytoplankton. The model simulates hunting and feeding behavior with exchange of energy during consumption (Table 2); agents tire and rest to regain energy; predators and prey exert extra energy when hunting. Predators change into a deeper color when feeding ((b) and (c)). Although the collective macro-level patterns cannot be categorized as emergent, the micro-level position of food in each species does influence the upper trophic level, resulting in a non-emergent grouping behavior.

4.5 N-tiered trophic network

The simulation in this section looks at various properties of complex systems using the model resolution in Section 3 in various scenarios. The systems simulate large-scale biotic and abiotic communication amongst agents of different species and the global and local environment.

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

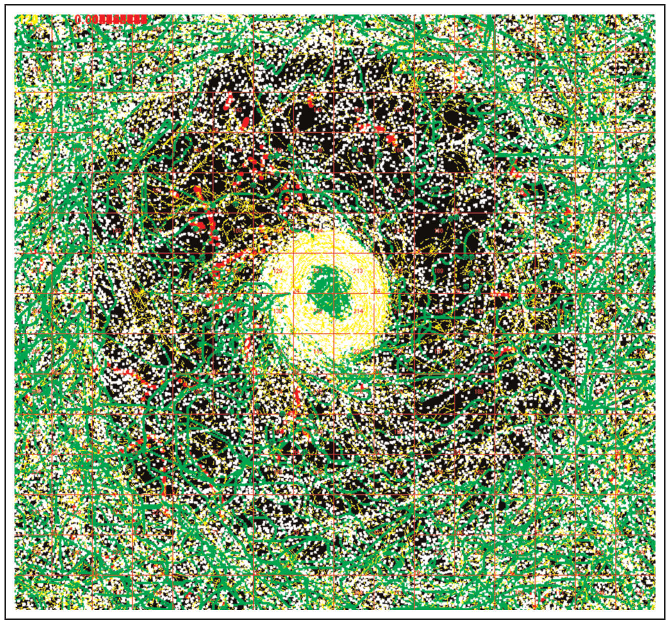

Figure 7. Local climate in a larger population of agents. One hundred and twenty agents avoiding the "wall" of heat emitters after 1621 simulation cycles.

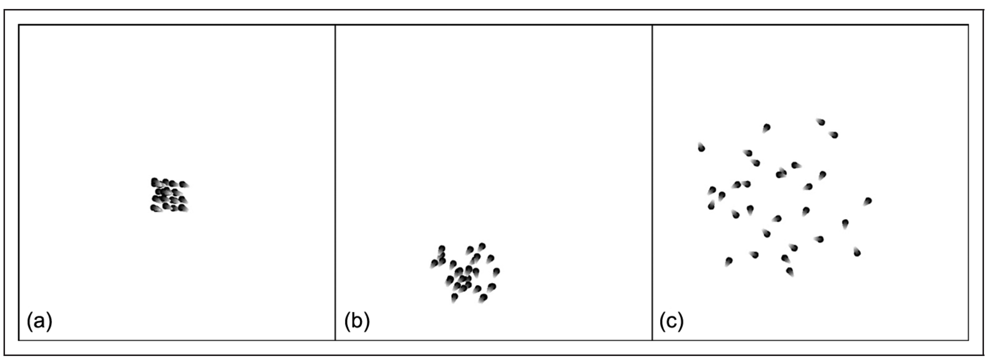

Figure 8. The common flocking behavior: (a) separation variable = 80; (b) separation variable = 20; (c) separation variable = 1.

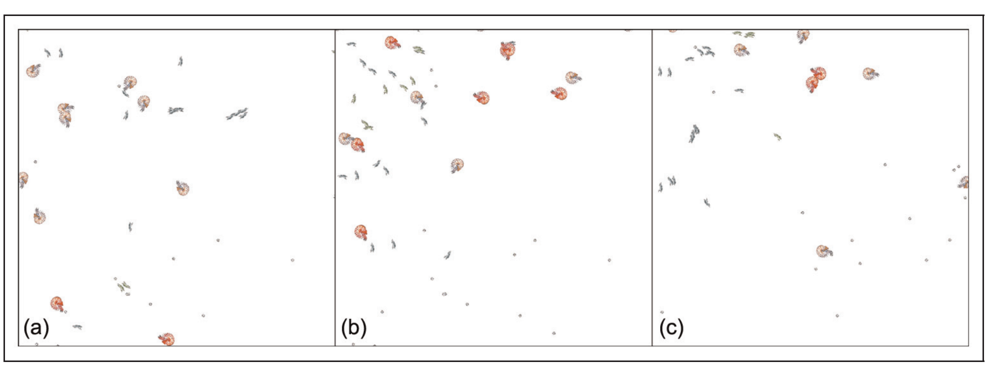

Figure 9. A simple trophic network with two consumers and a producer. Intense feeding behaviors are seen in (b) and (c). Predators change into a darker color when feeding.

Figure 10 shows a simulation consisting of five species of agents (tree, two herbivores, three carnivores). Soil condition patches and temperature emitters are placed on the four quadrants to observe effects when climate and condition information is communicated to the agents. The gradient patch of the soil in the first quadrant allows growth

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

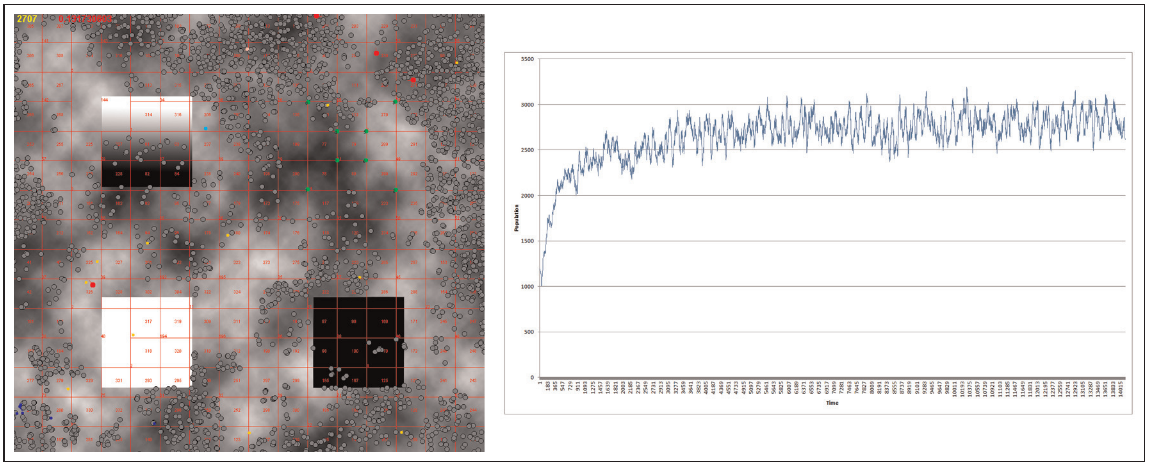

Figure 10. Global population stability (2500–3000) amongst five species of agents (a species of vegetation, two herbivores, and three carnivores). Soil conditions and temperature emitters are placed on the four quadrants to observe effects when climate and condition information is communicated to the agents. Plots are log scale.

only at the bottom half (darker patch), the white patch on the lower left quadrant prevents growth and the patch at the bottom right quadrant allows growth. The four temperature emitters on the upper right quadrant discourage vegetation from growing well in the area. The intensity map of the general terrain causes vegetation to settle on pockets of niches. The population of all species grows quickly and reaches a state of equilibrium between 2500 and 3000 agents, as agents interact in the trophic network. In this simulation, it can be seen that the micro states of the agents are coordinated by the space they occupy in competition with other agents, and the region that the agents inhabit is constrained by their adaptability to the local conditions caused by the macro-level effects of the soil: global temperature with local temperature emitters.

The simulation in Figure 11 shows the interaction amongst 11 species of agents spanning 10 trophic levels. The trophic chain is as follows – vegetation–herbivore1 and 2–carnivore1–carnivore2–carnivore3–carnivore4–carnivore5–carnivore6–carnivore7–carnivore8. Decline in any of the nodes causes a decline in the upper chain. The intensity map of the general terrain causes vegetation to settle on pockets of niches, which resulted in the movement patterns of consumers in the upper trophic chain. Exponential vegetation growth in the niches causes the growth of herbivores, which in turn causes the unhindered growth of carnivore1, as carnivore2 declines earlier. The decline of carnivore2 was due to its distant location in relation to carnivore1, set in the initial state of simulation. The simulation demonstrated the micro–macro integration where changes in the micro-level properties of each species of agents directly influence the upper trophic level. This cascading effect can clearly be seen in Figure 11.

The scenario in Figure 12 shows the population movement dynamics and exponential growth of herbivores due to the decline in population in the upper trophic chains, a macro–micro effect. The massive herbivore growth in pockets of niches where vegetation is abundant can be seen in Figure 13 (7–10). As vegetation is depleted in the niches, the population of herbivores declines. The herbivores first disperse (the concentration decreases) before dying of hunger. Figure 13 shows the interaction graph of the trophic network. The trophic chain effects can be observed. The figure shows interaction amongst 11 species in 10 trophic levels. The trophic chain is as follows–vegetation–herbivore1 and 2–carnivore1–carnivore2–carnivore3–carnivore4–carnivore5–carnivore6–carnivore7–carnivore8. The graph shows the decline in lower trophic chains resulting in the decline of upper trophic chains later in the simulation.

The general behavior of each agent is its survivability and the reproduction of progenies. The survival of the entire ecosystem depends on the balance of the organisms that inhabit the landscape. If a predator out-grows the prey, imbalance occurs and the system perishes. If the consumers out-grow the producers, food becomes scarce and the system is at a dilemma. When the environment is habitable, the population of producers multiplies and leads to the healthy population growth of the upper trophic level. The difficulty of such a model is the maintenance of equilibrium, but this is research for a model-focused paper. The micro–macro to micro link is clearly seen in this simulation.

5. Discussion

This article addresses the model resolution of agent preferences and behavior, their movement dynamics and

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

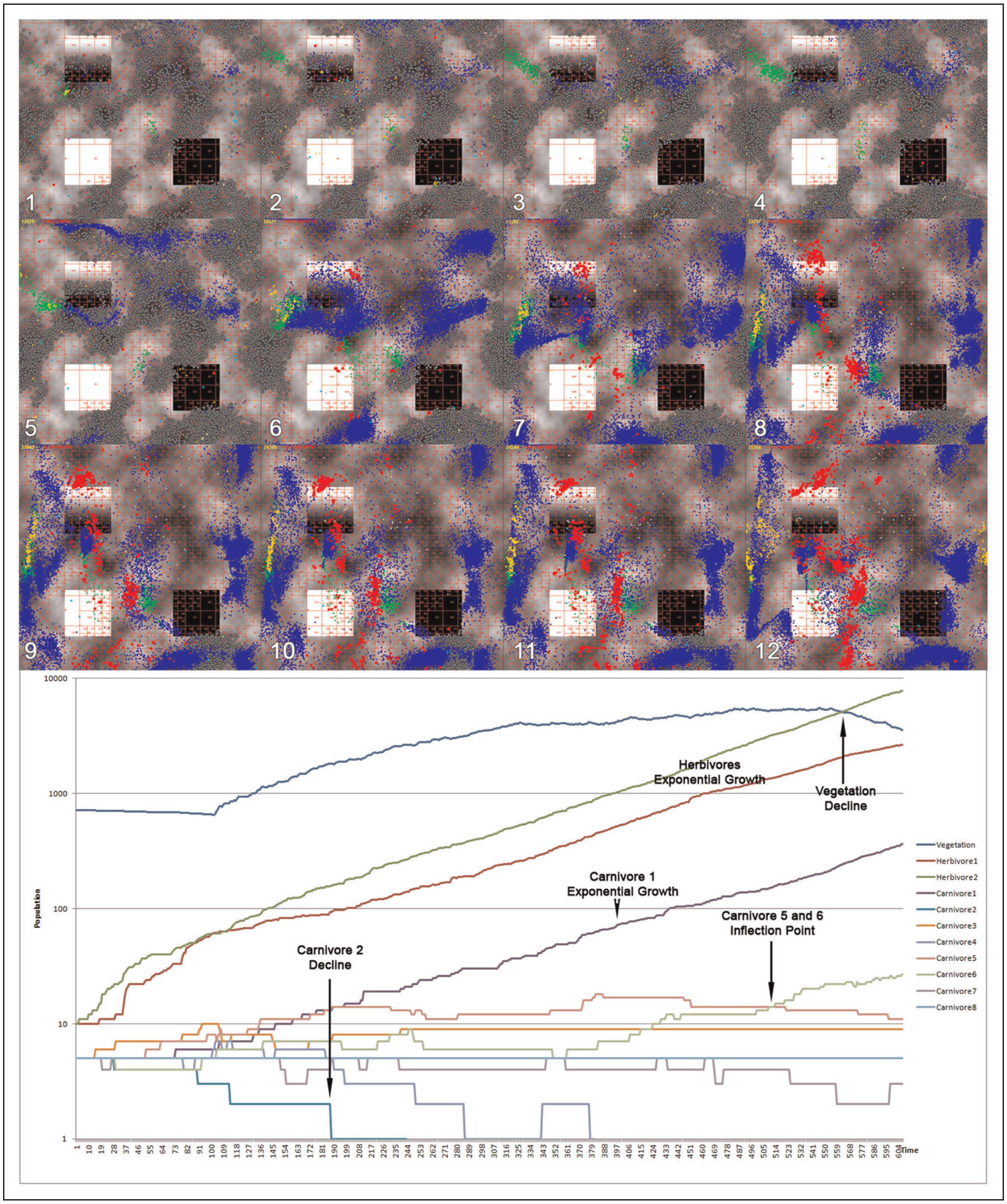

Figure 11. Interaction amongst 11 species in 10 trophic levels. The trophic chain is as follows: vegetation–herbivore1 and 2-carnivore1–carnivore2–carnivore3–carnivore4–carnivore5–carnivore6–carnivore7–carnivore8. Plots are log scale.

interaction in *n*-tiered networks, and how agent interactions and communications are computed within an optimized OO framework that facilitates efficiency and scalability. The OO framework presented in this article has the potential for simulating high-resolution complex systems composed of simple entities intertwined within a network of continuous information transfer between those entities. The hypothetical scenarios simulated here have shown that the communication of information or interaction of simple rules in agents locally can give rise to emergent behavior and self-organization, demonstrating the micro–macro link in the ecology of high-resolution models. In such systems (biological, ecological, cultural, social, economic, environmental, etc.), time, space and the

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

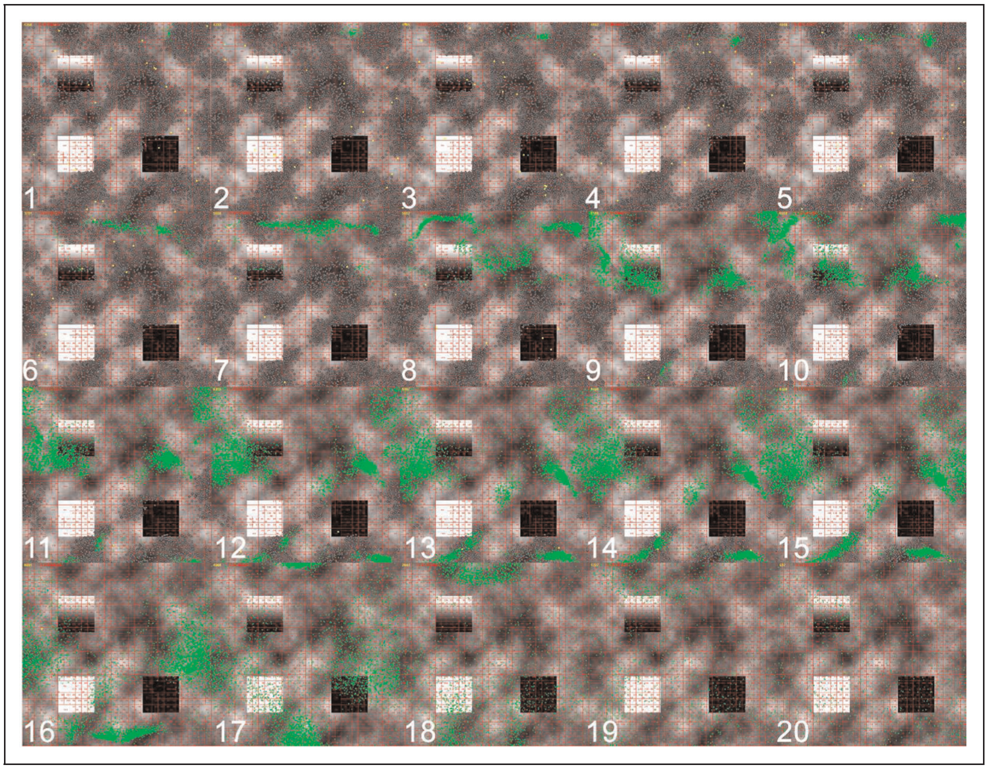

Figure 12. Population movement dynamics and exponential growth of herbivores due to the decline in population in the upper trophic levels.

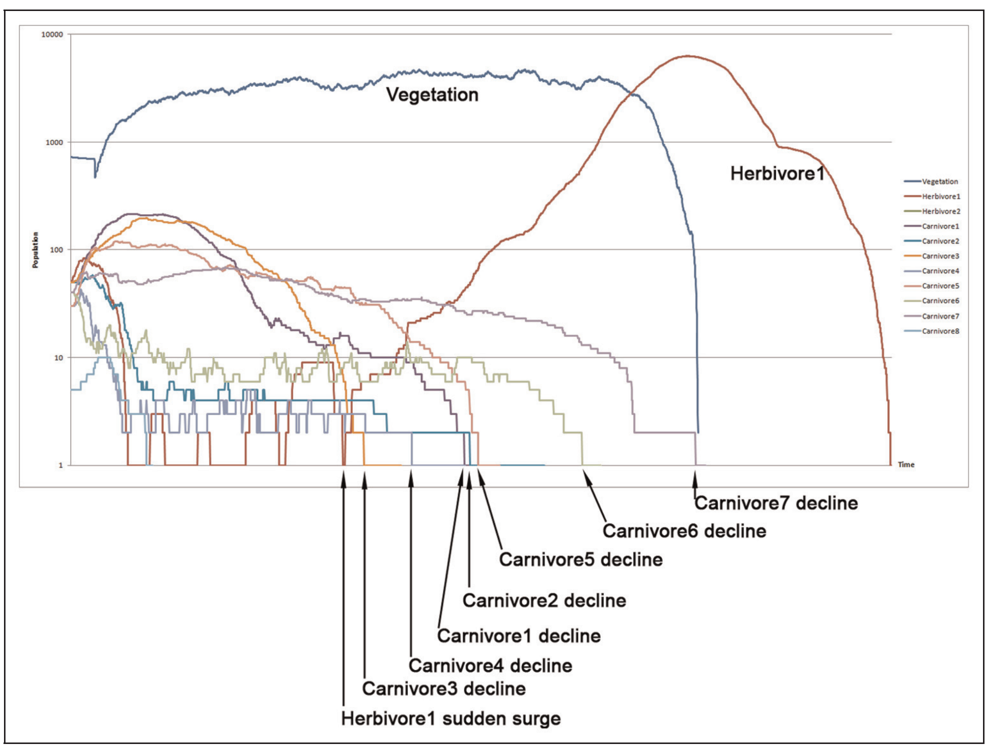

Figure 13. Trophic chain effect seen in a graph illustrating the interactions among 11 species of agents spanning 10 trophic levels. The trophic chain is as follows: vegetation–herbivore1 and 2–carnivore1–carnivore2–carnivore3–carnivore4–carnivore5–carnivore6–carnivore7–carnivore8. The graph shows the decline in lower trophic chains resulting in the decline of upper trophic chains later in the simulation. Plots are log scale.

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

context of the environment are important factors determining the forming of complex systems.

The computational modeling and simulation of complex systems requires a framework for supporting multiple resolutions in the diversity of agents, agent preferences and their information communication and complex interaction. By using ecological models as scenarios, the framework presented here demonstrated flexibility and scalability. Ecological models are the most complex, as they can potentially involve very large populations that increase and decrease in a very short burst of time. The algorithmic management of agents using the adaptive Quadtree is absolutely crucial, as the synchronization of agent interaction needs to be in the same simulation time steps, for example, predators need not interact with prey that have been captured as food. Ecological models can also span large spatial and temporal scales, involving agents in the microorganism category, such as soil chemistry, that have a direct link with terrestrial vegetation, which in turn interacts with the flora and fauna in the ecology. The range of agents navigating the landscapes is in very different resolution levels, for example, microorganisms' movements are measured in nanometers in comparison with arthropods' moving in millimeters. In such cases, modeling with current cellular automata-based simulation software can be very limiting – there can only be a single cell size at any given moment and the cell can only be occupied by an agent, or agents, in a stacked order. The free-roaming agents presented in this paper can be of any size and movement speed can be in very high resolution – floating point resolution in three-dimensional (3D) environments and pixel resolution in two-dimensional (2D) environments. With regards to temporal scales, the adaptive Quadtree optimization of entity interaction in logarithmic time means that simulation time increases at a manageable speed with the increase of agent population. Furthermore, the parallelization of the adaptive Quadtree algorithm in the near future will pave the way for greater performance scalability. Resolution in agent behavior is equally important, especially when diversity of agents is needed in the model. Such flexibility will ensure that a diverse complex system can be modeled and simulated spanning large spatial temporal scales. Since highly complex ecological models spanning unlimited trophic chains and interactions can be modeled and scaled based on the research presented here, many similar categories of complex systems can also be modeled and simulated.

The research presented here can be applied to a diverse set of complex systems with high dynamic population changes, deep trophic chains and large degrees of networked interaction. This implies that past and present models of the real world can be simulated, with potentials for predictive modeling of future scenarios. For example, archaeology reconstructs maps of past societies through proxy evidence. Such evidence of culture and the environments they live in provides resources for interpretations. These datasets could be used for modeling an archaeological site, with specific research questions either focusing on the settlement patterns of past societies based on land resource or on how past hunter–gatherer communities formed by generating data for social network analysis. Hundreds of simulation scenarios can be processed in order to test hypotheses. The framework can also be used for modern datasets and predictive modeling in areas as diverse as ecology, society, culture and climate change.

In this paper, the various challenges that accompany the modeling of complex systems have been demonstrated. While the focus of the paper is centered on the model resolution of complex systems, the importance and relatedness of these challenges to successful modeling of complex systems cannot be disregarded. These challenges are as follows.

1. SEs – the creation of enhanced Simulation Environments (eSEs). How can virtual environments incorporate micro and macro conditions, etc., that allow agents to react, contribute to and evolve in a way that is similar to real-world systems?
2. Large-scale agent simulation – the simulation of scalable population of real-world agents (biological, social, cultural, economic, environmental, etc.) using adaptive data structures and the segmentation of processes into manageable units for interaction and communication amongst agents and environments.
3. Model resolution – OO Modeling of the DPBICS of the system – object orientation efficiently reuses similar code patterns for simulating agent processes via inheritance and polymorphisms. What are the common DPBICS patterns found in the different aspects of simple and complex organisms that could be modeled in OO concepts?
4. Agent Networks and Interaction: Trophic Networks, Resource Usage, Social Relationships and Environmental Factors – these topics are highly intertwined and are programmatically difficult even in small-scale ecosystems (limited number of population, resources and relationships). Agents have to sense the environment and access all macro and micro factors, traverse the availability of resources at accessible proximity (there may be hundreds of thousands in computer memory) and deal with a multitude of social relationships via tagging (friend, foe, etc.). How can these networks and interactions can be managed in an efficient way?

The challenges laid out above have been addressed in this article, and examples of hypothetical scenarios from biology and ecology are provided to test the flexibility and

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

potentials of the approach presented here. Future work will incorporate time compression algorithms for simulating thousands to hundreds of thousands of years of complex system processes to a period of hours, days and months. Other aspects of research will require the parallelization of simulation processes using high-performance computing clusters – partitioning landscapes and distributing information related to the partitions to the clusters. These, together with future theoretical and empirical studies, will inform and reinforce our understanding of complex adaptive systems in our world.

Complex systems modeling and simulation is an exciting and promising field in the context of many scientific movements in this time and age. There is scope for research in this area for many years to come, which this article hopes to address initially.

Funding

This research received no specific grant from any funding agency in the public, commercial or not-for-profit sectors.

References

1. Manuel FE. *The religion of Isaac Newton*. Oxford, UK: Clarendon Press; 1974.
2. Oxford Dictionaries website. http://oxforddictionaries.com. Accessed October 16, 2010.
3. Aristotle. *Metaphysica*. 10f–1045a.
4. De Wolf T and Holvoet T. Emergence versus self-organisation: different concepts but promising when combined. *Lect Notes Comput Sci* 2005; 3464: 1–15.
5. Lewin R. *Complexity: life on the edge of chaos*. London: Phoenix, 1993.
6. Resnick M. *Turtles, termites, and traffic jams: explorations in massively parallel microworlds*. Cambridge, MA: MIT Press, 1994.
7. Ch'ng E. An artificial life-based vegetation modelling approach for biodiversity research. In: Chiong R (ed.) *Nature-inspired informatics for intelligent applications and knowledge discovery: implications in business, science and engineering*. Hershey, PA.: IGI Global, 2009.
8. Keller EF and Segel LA. Initiation of slime mold aggregation viewed as an instability. *J Theor Biol* 1970; 26: 399–415.
9. Simon HA. *The science of the artificial*. Cambridge, MA: MIT Press, 1997.
10. Ch'ng E. Modelling the adaptability of biological systems. *Open Cybern Systemic J* 2007; 1: 13–20.
11. Ch'ng E. Spatially realistic positioning of plants for virtual environments: simple biotic and abiotic interaction for populating terrains. *IEEE Comput Graphics Appl* 2011; 99: 66–77.
12. Ch'ng E, Stone RJ and Arvanitis TN. Evaluating artificial life-based vegetation dynamics in the context of a virtual reality representation of ancient landscapes. In: *virtual systems and multimedia*, Ghent, Belgium, 3–6 October 2005.
13. Grimm V and Railsback SF. *Individual-based modeling and ecology*. Princeton, NJ: Princeton University Press, 2005.
14. Huston M, Deangelis D and Post W. New computer models unify ecological theory. *Bioscience* 1988; 38: 682–692.
15. Breckling B, Müller F, Reuter H, et al. Emergent properties in individual-based ecological models – introducing case studies in an ecosystem research context. *Ecol Modell* 2005; 186: 376–388.
16. Pearson RG and Dawson TP. Predicting the impacts of climate change on the distribution of species: are bioclimate envelope models useful? *Global Ecol Biogeogr* 2003; 12: 361–371.
17. Grimm V. Ten years of individual-based modeling in ecology: what have we learned and what could we learn in the future? *Ecol Modell* 1999; 56: 221–224.
18. Ch'ng E. A behavioural agent model for synthesizing vegetation distribution patterns on 3D terrains. *Appl Artif Intell* 2009; 23: 78–102.
19. Reynolds C. Flocks, herds, and schools: a distributed behavioral model. In: *the proceedings of computer graphics (SIGGRAPH '87)*, 1987.
20. Huth A and Wissel C. The simulation of fish schools in comparison with experimental data. *Ecol Modell* 1994; 75: 135–145.
21. Kunz H and Hemelrijk CK. Artificial fish schools: collective effects of school size, body size and body form. *Artif Life* 2003; 9: 237–253.
22. Oboshi T, Kato S, Mutoh A, et al. Collective or scattering: evolving schooling behaviors to escape from predator. In: *artificial life VIII: proceedings of the fourth international workshop on the synthesis and simulation of living systems*, 2003.
23. Alexander JC and Giesen B. From reduction to linkage: the long view of the micro-macro debate. In: Alexander JC, Giesen B, Münch R, et al. (eds) *The micro-macro link*. Berkeley, CA: University of California Press, 1992.
24. Sawyer RK. Artificial societies. *Socio Meth Res* 2003; 31: 325–363.
25. Schillo M, Fischer K and Klein CT. The micro-macro link in dai and sociology. In: *proceedings of the second international workshop on multi-agent-based simulation-revised and additional papers*, Springer, 2001, pp. 133–148.
26. Drogoul A and Ferber J. Multi-agent simulation as a tool for modeling societies: application to social differentiation in ant colonies artificial social systems. *Lect Notes Comput Sci* 1994; 2–23.
27. Buskens V, Raub W and Van Assen M (eds). *Micro-macro links and microfoundations in sociology*. London, New York: Routledge/Taylor Francis, 2011.
28. Squazzoni F. The micro-macro link in social simulation. *Sociologica* 2008; 1. doi: 10.2383/26578.
29. Conte R and Castelfranchi C. Simulating multi-agent interdependencies. A two-way approach to the micro-macro link. In: *Social Science Microsimulation [Dagstuhl Seminar, May, 1995]*, Springer, 1996, pp. 394–415.
30. Setälä H, Marshall VG and Trofymow JA. Influence of micro- and macro-habitat factors on collembolan communities in Douglas-fir stumps during forest succession. *Appl Soil Ecol* 1995; 2: 227–242.
31. Balser T, McMahon K, Bart D, et al. Bridging the gap between micro - and macro-scale perspectives on the role of

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013

microbial communities in global change ecology. *Plant Soil* 2006; 289: 59–70.
32. Le QB, Park SJ and Vlek PLG. Land use dynamic simulator (Ludas): a multi-agent system model for simulating spatio-temporal dynamics of coupled human–landscape system: 2. scenario-based application for impact assessment of land-use policies. *Ecol Informat* 2010; 5: 203–221.
33. Parker DC, Manson SM, Janssen MA, et al. Multi-agent systems for the simulation of land-use and land-cover change: a review. *Ann Assoc Am Geogr* 2003; 93: 314–337.
34. Couclelis H. Cellular worlds: a framework for modeling micro - macro dynamics. *Environ Plann A* 1985; 17: 585–596.
35. Gernstein DR. To unpack micro and macro: link small with large and part with whole. In: Alexander JC, et al. (eds) *The micro-macro link*. Berkeley, CA: University of California Press, 1992.
36. Schelling TC. *Micromotives and macrobehavior*. New York, NY: WW Norton, 2006.
37. Nelson A. Some issues surrounding the reduction of macroeconomics to microeconomics. *Philos Sci* 1984: 573–94.
38. Morris A, Ross W, Hosseini H, et al. Modeling culture with complex, multidimensional, multiagent systems. In: Dignum V, Dignum F, Ferber J, Stratulat T (eds.) *Integrating cultures: Formal models and agent-based simulations*. Springer, 2012.
39. Guessoum Z. Adaptive agents and multiagent systems. *IEEE Distr Syst Online* 2004; 5: 1–4.
40. Nowak A and Latané B. Simulating the emergence of social order from individual behaviour. In: Gilbert N and Doran J (eds) *Simulating societies. The computer simulation of social phenomena*. London: UCL Press, 1994.
41. Bornhofen S and Lattaud C. Outlines of artificial life: a brief history of evolutionary individual based models. *Lect Notes Comput Sci* 2006; 3871: 226–237.
42. Langton CG (ed.) *Artificial life: an overview*. Cambridge, MA: MIT Press, 1995.
43. Ch'ng E. An efficient segmentation algorithm for entity interaction. *Biodiversity Informatics* 2009; 6: 5–17.
44. Schulz R and Reggia JA. Predicting nearest agent distances in artificial worlds. *Artif Life* 2002; 8: 247–264.
45. Snyder JM and Barr AH. Ray tracing complex models containing surface tessellations. In: *SIGGRAPH '87 proceedings*, 1987, pp.1–10.
46. Deussen O, Hanrahan P, Lintermann B, et al. Realistic modeling and rendering of plant ecosystems. In: *proceedings of SIGGRAPH '98 annual conference series*, 1998.
47. Soler C, XSillion FX, Blaise F, et al. An efficient instantiation algorithm for simulating radiant energy transfer in plant models. *ACM Trans Graph* 2003; 11: 204–233.
48. Aho AV, Hopcroft JE and Ullman JD. *The design and analysis of computer algorithms*. Reading, MA: Addison-Wesley, 1974.
49. Samet H. The Quadtree and related hierarchical data structures. *Comput Surv ACM* 1984; 16: 187–260.
50. Ch'ng E, Chapman H, Gaffney C, et al. Landscapes without figures: big data, long waves and the formative role of archaeological computing. *IEEE Comput* 2011; 44: 40–46.
51. Gaffney VL, Fitch S and Smith D. *Europe's lost world, the rediscovery of Doggerland (Cba Research Reports)*. York: Council for British Archaeology, 2009.
52. Grimm V, Berger U, Bastiansen F, et al. A standard protocol for describing individual-based and agent-based models. *Ecol Modell* 2006; 198: 115–126.
53. Ch'ng E. Macro and micro environment for diversity of behaviour in artificial life simulation. In: *artificial life session, the 6th international conference on soft computing and intelligent systems, the 13th international symposium on advanced intelligent systems*, Kobe, Japan, 20–24 November 2012.

Author biography

Dr Eugene Ch'ng is Senior Lecturer at the University of Birmingham. He is Innovations Director at the IBM Visual and Spatial Technology Centre and Do.Collaboration. He has formal education in a wide variety of fields (fine arts, graphic design, interior architecture, computer science and, electronic, electrical and computer engineering). His research has an overarching theme in complexity science. He specializes in complex systems, agent-based modeling, MASs and enhanced SEs that require large computing clusters for processing of agent–environment interaction and computer graphics rendering. He is involved in editorial boards, grant review panels, technical and program committees in international journals and conferences in his field.

Downloaded from sim.sagepub.com at Bibliotheek TU Delft on September 12, 2013