



A general overview of formal languages for individual-based modelling of ecosystems

Mauricio Toro

Universidad Eafit, Colombia



ARTICLE INFO

Article history:

Received 13 July 2018

Received in revised form 12 January 2019

Accepted 28 January 2019

Available online 5 February 2019

Keywords:

p-systems

Process calculi

Individual-based modelling

Formal modelling languages

Ecosystems

ABSTRACT

Various formal languages have been proposed in the literature for the individual-based modelling of ecological systems. These languages differ in their treatment of time and space. Each modelling language offers a distinct view and techniques for analyzing systems. Most of the languages are based on process calculi or P systems. In this article, we present a general overview of the existing modelling languages based on process calculi. We also discuss, briefly, other approaches such as P systems, cellular automata and Petri nets. Finally, we show advantages and disadvantages of these modelling languages and we propose some future research directions.

© 2019 Elsevier Inc. All rights reserved.

1. Introduction

The collective evolution of a group of individuals is of importance in many fields; for instance, in system biology, ecology and epidemiology. When modelling such systems, we want to know the emergent behavior of the whole population given a description of the low-level interactions of the individuals in the system. As an example, in *eco-epidemiology* the focus is on the evolution in time of the number of individuals infected in a certain population and how a small number of individuals infected may lead to an epidemic.

Eco-epidemiology can be seen as a particular case of *population ecology*. The main aim of population ecology is to gain a better understanding of population dynamics and make predictions about how populations will evolve and how they will respond to specific management schemes. In epidemiology, such management schemes can be a cure to a disease, mechanisms to prevent a disease such as vaccines, or mechanisms to prevent the *vector* (species infected with a disease) to spread a disease. To achieve these goals, scientists construct models of ecosystems and management schemes (e.g., [74]).

Luisa Vissat et al. argue that we can use these models to predict the system behavior and use this predictive power to support decision-making [43]. As an example, in the case of the spread of diseases, models are used to find the optimal control strategies for containing the spread or to predict the results of a population reduction, as a key way to control disease in livestock and wildlife [57].

Various formalisms have been proposed in the literature for the individual-based modelling of ecological systems. These formalisms differ in their treatment of time and space and each modelling language offers a distinct view and techniques for analyzing systems. Surprisingly, many formalisms developed for population modelling have been also used to model music interaction in real life performance; for instance process calculi has been used to model real-time performance in music [72], computer music improvisation [68] and interactive music scores [69].

E-mail address: mtorobe@eafit.edu.co.

In what follows, we explain, in Section 2, classic process calculi and different extensions of classic process calculi used to model population systems and some applications in ecology; in Section 3, P systems and some extensions used to model population systems; and, in Section 4, other approaches used for the formal modelling of population systems. Finally, in Section 5, we discuss the advantages and disadvantages of these formalisms and we recommend some future research directions in this field. Note that the formalisms explained in this article are not only strictly related to applications in ecology. Furthermore, this paper focuses on process calculi and P systems, but we also present other approaches for formal modelling of population systems such as cellular automata and Petri nets.

2. Process calculi

Process calculi (or process algebras) are a diverse family of related approaches for formally modelling concurrent systems. Process calculi provide a tool for the high-level description of interactions, communications, and synchronizations between a collection of independent processes. Process calculi also provide algebraic laws that allow process descriptions to be manipulated and analyzed, and allow formal reasoning about equivalences between processes.

In what follows, first we explain classic process calculi. After that, we explain, in more detail, different extensions of classic process calculi used to model population systems and some applications in ecology. Note that process calculi explained in this section are not only strictly related to applications in ecology.

2.1. Classic process calculi

CCS. The *calculus of communicating systems* (ccs) [47] is a process calculus introduced by Robin Milner around 1980. The calculus includes primitives for describing parallel composition, choice between actions and scope restriction. CCS is useful for evaluating qualitative correctness properties of a system such as deadlock or livelock. The expressions of the language are interpreted as a *labeled transition system*. Between two labeled transition systems, *bisimilarity* is used as a semantic equivalence. In fact, there is a tool to check bisimulation, as well as for the simulation and model checking of CCS and sCCS.¹

sCCS. Soon after the creation of CCS, Milner developed an extension of CCS. In *synchronous CCS* (sCCS) [48], concurrent processes proceed, in contrast to CCS, simultaneously in lockstep. This means that at every step the concurrent processes in a composite system perform a single action.

CSP. *Communicating sequential processes* (CSP) [36] is a process calculus first described in 1978 by C.A.R. Hoare. CSP is different from CCS: In CSP, processes composed in parallel cooperate in a multi-party synchronization. In contrast, in CCS, parallel processes interact with each other by means of a two-party synchronization. Another difference between CSP and CCS is that synchronization in CSP is made by symmetrical actions, whereas in CCS there are actions and co-actions. Soon after the invention of CCS and CSP, the need of sending channel names to channels themselves motivated the invention of the π -calculus.

π -calculus. The *π -calculus* (or π -calculus) [49] is a process calculus developed by Robin Milner, Joachim Parrow and David Walker in 1992, as a continuation of Milner's work on CCS. The π -calculus allows channel names to be communicated along the channels themselves. Using this feature, one is able to describe concurrent computations whose network configuration may change during the computation.

CCP. *Concurrent constraint programming* (CCP) was developed by Vijay Saraswat in 1991. In CCP, a system is modeled in terms of processes adding to a common *constraint store* partial information on the value of variables. Concurrent processes synchronize by blocking until a piece of information can be deduced from the store. CCP is based on the idea of a *constraint system* (CS). A constraint system is composed of a set of (basic) constraints and an entailment relation specifying constraints that can be deduced from others.

2.2. Stochastic process calculi

Stochastic process algebras aim to combine two successful approaches of modelling: *labeled transition systems* and *continuous-time Markov chains* [29]. Indeed, they argued that labeled transition systems are a very convenient framework to provide compositional semantics of concurrent languages and to prove qualitative properties. Markov chains, instead, have been used in performance evaluation and quantitative properties. The common feature of most stochastic process calculi is that their actions are enriched with rates of exponentially distributed random variables characterizing their mean duration.

¹ <http://homepages.inf.ed.ac.uk/perdita/cwb/doc/>.

Continuous pi-calculus. Ever since the invention of the pi-calculus, there has been an interest to use it to model ecosystems. A successful development on this direction is an extension of the pi-calculus with continuous time. The *continuous pi-calculus* [41] is an extension of the pi-calculus in which the processes evolve in continuous time. It can be seen as modular way to generate ordinary differential equations for a system behavior over time. The continuous pi-calculus has been used to model biochemical systems [41].

Stochastic ccs. In *stochastic ccs* (stoccs), output actions are equipped with a parameter characterizing a random variable with a negative exponential distribution, modelling the duration of the action [29]. Input actions are denoted with a weight, a positive integer used to determine the probability that the specific input is selected when a complementary action is executed [29]. There are no tools available for their simulation nor verification up to our knowledge. This calculus has not been used to model ecosystems, but the definition of *rate-based transition systems* has inspired many other modelling languages [29].

As argued by Cardelli et al. in [21], one cannot guarantee associativity of parallel composition operator up to stochastic bisimilarity when the synchronization paradigm of ccs is used in combination with the synchronization rate computation based on apparent rates. This is a problem, specially, in presence of dynamic process creation.

Stochastic pi-calculus. The *stochastic pi-calculus* is an extension of the pi-calculus with rates associated to the actions, developed by Corrado Priami in 1995. Recently, Cardelli et al. developed new semantics with replication and fresh name quantification [21]. Cardelli et al. argued that parallel composition failed to be associative up to stochastic bisimilarity with previous semantics. With the new semantics it is now possible to capture associativity, which will lead to new applications and simulation tools. In fact, there is already a simulation tool for the stochastic pi-calculus named the *stochastic pi machine* (*Spim*)²; Spim uses the probabilistic model checker PRISM. Another approach for model checking has been developed by Norman et al. [50].

Stochastic pi-calculus has been used in system biology to model regulating gene expression by positive feedback [56].

PEPA. The properties that may be checked for concurrent systems modeled by an algebraic description include freedom from deadlock and algebraic equivalence under observation [33]. Hillston et al. argue that there are further properties of interest for process calculi models such as steady-state probabilities and rewards for performance measures. *Performance evaluation process algebra* (PEPA) [33] was defined by Hillston et al., in 1994, to allow the specification and verification of such properties. Every activity in PEPA has an associated duration which is a random variable with an exponential distribution, determined by a single real-number parameter. A PEPA process can be translated into a continuous-time Markov process [33].

PEPA has been used in system biology to model the *raf kinase inhibitor protein* on the extracellular signal regulated kinase signalling pathway [17] and to model measles epidemics in the UK from 1944–1964 [9].

Bio-PEPA. Bio-PEPA [25] is a process calculus for modelling and analyzing biochemical networks. Bio-PEPA is a modification of PEPA adding some features such as support for *stoichiometry* and general *kinetic laws*. Kinetic laws are functions which allow us to derive the rate of reactions from varying parameters such as the rate coefficients and concentration of the reactants [51]. According to Hillston et al., the main difficulty with PEPA is the definition of stoichiometric coefficients [25]: These coefficients are used to show the quantitative relationships of the reactants and products in biochemical reactions. A major feature of Bio-PEPA is the possibility to represent explicitly some features of biochemical models such as stoichiometry and the role of species in a given reaction. Bio-PEPA is also enriched with some notions of equivalence such as isomorphism and bisimulation, extended from PEPA. Bio-PEPA has been used in epidemiology to model the *H5N1 avian influenza* [26].

Stochastic CCP. Stochastic CCP (sccp) was developed by Bortolussi [16], in 2006. sccp is an extension of CCP by adding a stochastic duration to all instructions interacting with the constraint store. In sccp, each instruction has an associated random variable, exponentially distributed [16]. Bortolussi et al. also proposed an approximation of sccp semantics into ordinary differential equations [14].

Bortolussi et al. have used sccp to model bio-mechanical reactions such as an *enzymatic reaction* [16]. sccp has also been in system biology [15] and to model prey-predator dynamics [14].

2.3. Probabilistic process calculi

Process calculi depend on concepts of non-deterministic choice. Tofts argues that it is natural to consider extending such systems by adding a probabilistic quantification for non-determinism [67]. In terms of expressiveness of modelling, McCaig et al. argue that probabilistic choice is a more natural way to express individual behavior than stochastic rates of activities

² <http://research.microsoft.com/en-us/projects/spim/>.

for epidemiological models [45]. Another alternative for semantics of probabilistic calculi is having both non-deterministic and probabilistic choice, as explained by Norman et al. [50].

Weighted SCCS. A probabilistic calculus, derived from Milner's sccs [48], is *weighted sccs* (wsccs) [67]. This calculus was developed by Chris Tofts, in 1994. wsccs is synchronous because the purpose is to quantify the relative frequency of free simultaneous choice. In an asynchronous system, choices may be resolved at arbitrary times, thus giving a choice which may not be between equally free objects. wsccs does not quantify choices directly with probabilities, but it uses weights: Weights are interpreted as probabilities via the concept of relative frequency.

McCaig et al. presented a new semantics in terms of *mean field equations* in [45]. This semantics captures the average behavior of the system over time, without computing the entire state space; therefore, avoiding the state-space explosion problem. The new semantics is shown to be equivalent to the standard discrete-time Markov chain semantics of wsccs, as the number of processes tends to infinity. Using wsccs's *probabilistic workbook*, one can analyze systems in wsccs [45]. There is also some work on population models for stochastic process algebras done by Hillston et al. [35,73]. A related interesting work is the model checking of mean-field approximations of continuous-time Markov chains [13].

Note that this calculus has been used to model human-population growth [45] and wsccs has been employed in various ecological studies by its author and others [65,44].

Probabilistic CSP#. *Hierarchical probabilistic CSP* (PCSP#) [66] extends CSP with probabilistic choice and data structures. PCSP# was developed by Jun Sun et al. in 2010. PCSP# combines sequential programs defined in a simple imperative language such as C# with high-level specifications such as parallel composition, choice and hiding, as well as probabilistic choice. Its underlying semantics is based on *Markov decision processes*.

Although PCSP#'s semantics are given on Markov decision processes, Sun et al. argue that existing probabilistic model checkers have been designed for simple systems without hierarchy [66]. For that reason, they extended an existing model checker to support probabilistic model checking of hierarchical complex systems. Probabilistic CSP has been used to model sensor networks, security protocols and probabilistic algorithms. PCSP# has not been used to model ecosystems.

Probabilistic pi-calculus. The *probabilistic pi-calculus* was developed by Herescu et al. in 2000 [34]. In the probabilistic pi-calculus, there are two types of choice operators: non-deterministic and probabilistic. The operational semantics for probabilistic extensions of the pi-calculus are typically expressed in terms of Markov decision processes. Existing semantics are concrete, which means processes are encoded directly into Markov decision process. Norman et al., defined a symbolic representation of the calculus semantics [50]. The main feature of Norman et al.'s symbolic semantics is that the input variable of input transitions is kept as a name variable and the communication rule that matches between the input and the output channel is represented by a condition [50]. Norman et al. used those semantics for model checking of the calculus using probabilistic model checker PRISM. Norman et al. also used this calculus to model and verify randomized security protocols [50]. Probabilistic pi-calculus has also been used for model checking models of system biology [40].

Probabilistic CCP. There is a timed, probabilistic, non-deterministic extension of CCP (PNTCC) [52], developed by Pérez et al. in 2009. They defined the semantics as a discrete-time Markov chain in which non-determinism is unobservable because it is removed by a scheduler that must be defined along with the model. PNTCC has not been used for models in system biology nor ecology.

2.4. Spatial process calculi

Spatial aspects and locations are required in ecosystems because such systems can be physically distributed in space and this distribution can affect the length of time required for an activity to occur [31]. Spatial process calculi are extensions of process calculi with different notions of space. Spatial features of biological systems can be studied in different ways [51]. Pardini argues that most computer-science formalisms focus on the abstract spatial feature which concerns the *modelling of compartments*; nonetheless, he also argues that there are other formalisms that focus on a more *concrete notion of space* (e.g., continuous 2D space) [51].

PALPS. *Process algebra with locations for population systems* (PALPS) can be considered as an extension of CCS with probabilistic choice, and with locations and location attributes [55]. It shares a similar treatment of locations with process algebras developed for reasoning about mobile ad hoc networks; for instance, Galpin's spatial extension to PEPA and Kouzapas et al.'s calculus for dynamic networks [39]. Nonetheless, such approaches are stochastic. PALPS shares some similarities with sccs although it does not provide synchronous parallel composition nor probabilistic choice based on weights. PALPS considers a two-dimensional space where locations and their interconnections are modeled as a graph upon which individuals may move as computation proceeds. Compartments are static in PALPS. In this matter, PALPS is related to the spatial calculi previously described such as the calculus of wrapped compartments or to cellular automata. PALPS was used to model prey-predator dynamics [55].

PALPS was also extended with *process ordering* because simulations carried out by ecologists often impose an order on the events that may take place within a model [54]. As an example, if we consider mortality and reproduction within a

single-species model, three cases exist: concurrent ordering, reproduction preceding mortality and reproduction following mortality. Process ordering in PALPS was used to model the reproduction of the *parasitic varroa mite* that attacks honey bees [54].

Synchronous PALPS (S-PALPS). To alleviate the problem of the interleaving nature of parallel composition and high level of non-determinism in PALPS, Toro et al. proposed a new semantics of PALPS, and an associated PRISM translation, that disassociates the number of modules from the maximum number of individuals, while capturing more faithfully the synchronous evolution of populations and removing as much unnecessary non-determinism as possible. The synchronous extension of PALPS is named *synchronous PALPS (S-PALPS)* [71]. This synchronous semantics of PALPS implements the concept of *maximum parallelism*: at any given time all individuals that may execute an action will do so simultaneously.

Toro et al. developed a mean-field semantics for S-PALPS [70]. This semantics allows an interpretation of the average behavior of a systems as a set of recurrence equations. Recurrence equations are an approximation useful when dealing with a large number of individuals, as it is the case of epidemiological studies. S-PALPS was used to model dengue transmission in Bello (Antioquia), Colombia, in particular, to analyze the factors involved in the epidemiology of dengue disease [70]. The model presented by Toro et al. in [70] is an individual-based version of the model presented in [74].

Spatial PEPA. Galpin introduced spatial notions to the stochastic process algebra PEPA, creating *spatial PEPA* [31], in 2009. Galpin was motivated by both computer networks and epidemiology where locations of actions or processes may affect the time taken by an event. She defined a very general stochastic process algebra with locations, where locations are introduced to both actions and processes.

Galpin defined a weighted hypergraph structure to represent locations. The locations may or may not have structure. Nodes in a directed graph may represent computers in a network or locations in n-dimensional space. The graph edges could represent simple network connectivity, or a tree structure representing nesting of locations. As another example of its generality, the weight in the hypergraph structure could be obtained from distance function or from something else.

Galpin used Spatial PEPA to model single packet traversal through network [31]. Spatial PEPA has not been used to model ecosystems.

BioAmbients. *BioAmbients* [61] calculus was developed by Cardelli et al. in 2004. It was motivated by the Ambient calculus [63], originally developed by Cardelli for the specification of process location and movement through computational domains [20]. Bio-molecular systems, composed of networks of proteins, underlie the major functions of living cells. Compartments are the key to organization of such systems, according to Cardelli et al., BioAmbients is inspired in such biological compartments. The novelties with respect to the Ambient calculus are the movement of molecules between compartments, the dynamic rearrangement of cellular compartments and the interaction between molecules in a compartmentalized setting. Cardelli et al. adapted the *BioSpi* simulation tool for their new calculus and used it to study a complex multi-cellular system [61].

Brane calculus. A biological cellular membrane is an oriented closed surface that can perform various molecular functions. One constraint of such membranes is bi-tonality. This constraint requires nested membranes to have opposite orientations, so they can be coded with a coloring system of two tones. The organization of these membranes inspired the creation of the *Brane calculus*, in 2004, by Cardelli et al. [18]. In Brane, there is composition of systems, composition of membranes and replication. There is also a choice operation that can be added to membranes and communication on CCS style. Cardelli argues that it is difficult to encode this calculus on BioAmbients although they are closely related. The Brane calculus has been used to model the behavior of the *semliki forest virus* [46].

Calculus of wrapped compartments. The *calculus of wrapped compartments (cwc)* [12] is also based on the notion of a compartment. cwc has no explicit structure modelling a spatial geometry; nonetheless, Bioglio et al. argue that the compartment labeling feature can be exploited to model various examples of spatial interactions in a natural way [12].

To model 2D-space, cwc allows to label a compartment as a 2D-coordinate, denoted by a tuple of integers (row, column), thus they can define a two-dimensional grid [12]. Modelling the space as a two-dimensional grid, this calculus was used to analyze the growth model for *arbuscular mycorrhizal fungi* [12]. Biochemical transformations are described via a set of stochastic reduction rules which characterize the behavior of the represented system. Stochastic simulation can be defined along the lines of Gillespie's simulation. Bioglio et al. performed simulations of the growth model using the *surface language* (a language they developed) [12].

Spatial pi-calculus. Most calculi have focused on population-based approaches or on how to place and move individuals relative to each others position in space. *SpacePi* [37] extends the pi-calculus with continuous time and space. Processes are embedded into a vector space and move individually. Only processes that are sufficiently close can communicate.

According to John et al., existing spatial individual-based approaches have so far focused on indirect, relative space [37]. For that reason, John et al. developed SpacePi to explicitly model location and movement of individuals in absolute space. By combining an individual-based perception and absolute space, not only diffusion processes, but also active transportation

inside membranes can be modeled realistically at different resolutions [37]. This makes their approach different from other pi-calculus extensions, like the stochastic pi-calculus. This implies that existing simulation engines based on Gillespie's algorithm cannot be used; this also implies, that simulation will be less efficient than the one obtained with engines based on Gillespie's algorithm. This calculus has been used by John et al. to model the behavior of a euglena, a micro-organism living in inland water which motion is influenced by sun irradiation. John et al. ran simulations on a tool of their own [37].

Spatial CCP. Barco et al. recently proposed a spatial extension of CCP [5]. In spatial CCP, each process has a computation space that can be shared with other processes. In addition, it is possible to have nested computation spaces. Spatial CCP has not been used to model ecosystems.

Modelling in Ecology with Location Attributes. *Modelling in Ecology with Location Attributes* (MELA) is a process calculus to formally describe ecological systems, with focus on space abstraction and environmental description [43]. Like other stochastic process algebras, MELA allows complementary analysis techniques, such as stochastic simulations, numerical solution of ordinary differential equations. In MELA, the modeller can choose between different spatial structures; for instance, a graph, a discrete line segment, a 2D grid, a 3D grid, or a nested spatial structure. MELA has been used to model prey-predator dynamics (the Lotka–Volterra model) and the spread of a Cholera epidemic [43]. A disadvantage of MELA is the lack of tools to support the spatial and temporal analysis that could be possible using its semantics.

Process Algebra for Located Markovian Agents. *Process Algebra for Located Markovian Agents* (PALOMA) [30] uses multi-class multi-message Markovian agents to model collective systems comprised of populations of agents which are spatially distributed. PALOMA is equipped with both discrete event and differential semantics. The discrete semantics provides the theoretical foundation for discrete event simulation and the differential semantics allows us to automatically derive the underlying mean-field model. PALOMA was used to model a simplified scenario of the 1918–1919 flu epidemic in central Canada and to investigate the effect of quarantine on the spread of the flu.

Other extensions of CCS. Bartocci et al. proposed an extension of CCS to model 3D space. This extension is called the *shape calculus* [7,6].

Other extensions of the Pi-calculus. Pardini described several spatial extensions of the Pi-calculus [51]: He mentioned *Beta binders*, $\pi@$ calculus, *attributed π -calculus* ($\pi@$ calculus), the *imperative pi-calculus* and the 3π calculus. In Beta binders, the notion of compartments is represented by the means of *boxes* which contain pi-calculus processes. In $\pi@$ calculus, channels are associated with compartments. In attributed pi-calculus, processes can have attributes, for instance, information of their location. In imperative pi-calculus, there is a global store with information on the volume of each compartment. Finally, in the 3π calculus, processes are embedded in 3D space. It is worth to note that some of these extensions have been also extended with stochasticity, as it is the case of the $\pi@$ calculus.

Beta binders has been used to model biological interactions [58] and the $\pi@$ calculus has been used to model the *euglena's phototaxis*, gene regulation at *lambda switch* and population models [38].

3. P systems

A different approach towards modelling of ecological systems is that of *P systems* [59,60]. *P system* [59] is a class of distributed and parallel computing described by Gheorghe Păun in 2001. Each *membrane* represents a region and contains a multiset of objects. A *cell* is considered, in an abstract way, a hierarchy of compartments enclosed by membranes. Each compartment may include elementary objects as well as other compartments. Processes in a cell are viewed as sequences of discrete events.

Romero et al. argue that P systems started from the assumption that the processes taking place in the compartmental structure of a living cell can be interpreted as computations [62]. Romero et al. argue that a P system consists of cell-like membrane structures (known as compartments) in which one places multisets of objects that evolve according to given rewriting rules (i.e., evolution rules).

3.1. Stochastic P systems

Spicher et al. described a sequential strategy for executing P systems based on Gillespie's stochastic simulation algorithm [64]. They defined a stochastic simulation of a new class of P systems. This new kind of P systems are called *stochastic P systems*. In rewriting systems, the policy for deciding which rule(s) will be applied is called the application strategy [64]. They argued that in case of conflicts, rules are selected non-deterministically. Maximal-parallel strategy is well suited for the modelling of discrete dynamic systems operating synchronously. This strategy means that the rules are applied simultaneously in a maximal way during each rewriting step. This strategy is less suited to capture events that occur asynchronously in continuous time.

Gillespie's algorithm is an alternative rule application strategy for P systems. Gillespie's algorithm leads to a sequential application strategy for these rules: only one rule is applied in each derivation (simulation) step [64].

Spicher et al. implemented stochastic P systems using the spatially-explicit programming language MGS [64]. Spicher et al. have used stochastic P systems to model the *Lotka–Volterra auto-catalytic system* and the life cycle of the *similiki forest virus* [64]. Romero et al., provide a translation of stochastic P systems to PRISM, transforming a P system specification into a PRISM input [62]. Romero et al.'s translation allows model checking of stochastic P systems.

3.2. Probabilistic P systems

Probabilistic P systems have been applied to a wide range of situations in the field of ecology (e.g., [11]). The semantics of probabilistic P systems is closely related to s-PALPS: rules are usually applied with maximal parallelism while several proposals have been considered on resolving the non-determinism that may arise when more than one combination of rules is possible, for instance, [53,22]. Probabilistic P systems have been applied to model the population dynamics of various ecosystems [10,8] as well as to study the invasiveness of eastern species in European water frog populations [2]. As an example, there is a model of mosquitoes in Italy that uses attributes such as temperature and humidity [8].

In *probabilistic P systems*, the rules include a real number between zero and one: a probability of the rule for its application [1]. As expected, the sum of all rules with the same left-hand side of a rule must be one. The rules are applied in a maximal-consistent parallel way, as usual. The inherent randomness and uncertainty in ecosystems is captured by using probabilistic strategies.

Another application of probabilistic P systems in ecology is to model an ecosystem of some scavenger birds [1]. Cardona et al. described a P system for modelling an ecosystem based on three vulture species and the prey species from which vulture species obtain most of their energy requirements [1]. They include rules to limit the maximum amount of animals that can be supported by the ecosystem as well as the amount of grass available for the herbivorous species. They consider two regions: the skin and the inner membrane. The first region controls the densities of every species do not overcome the threshold of the ecosystem. The skin is where the animals reproduce and the inner membrane where they feed or die. Note that this model is not spatially explicit. Cardona et al. simulated their ecosystem model using the specification language *P-Lingua* for P systems [1].

3.3. Other extensions of P systems

There are at least two extensions to probabilistic P systems: *Dynamical probabilistic P systems* [23] and *multi-environment P systems* [51]. In both extensions, probabilities are associated with rules and the values may vary through time. A multi-environment P system model is composed of a set of environment, each containing a probabilistic P system. All the P systems inside the environments share some common features such as the alphabet of symbols, membrane structure and evolution rules. Multi-environment P systems have been used to model a simple (but realistic enough) ecosystem where a carnivore and several herbivorous species interact [27].

Another class of term rewriting formalism, similar to P systems, is the *calculus of looping sequences* (CLS) that allows to represent membranes explicitly with the calculus syntax. By the means of rewriting rules, it is possible to create, dissolve and merge membranes [51]. Recently, Pardini et al. extended this calculus with spatial information [4].

4. Other approaches

There exists a variety of other proposals which introduce locations or compartments into formal frameworks. In what follows we explain some of them.

Processes in space. This modelling language [19] has been used to analyze the influence of network topologies on local and global dynamics of metapopulation systems [11].

Cellular automata. *Cellular automata* were proposed by Von Neumann, in 1949. There are one-, two- and three-dimensional cellular automata. Cellular automata schematize the space into a regular lattice, according to the spatial scale of the system studied. Each cell has some properties and takes a value from a finite set of states. The value is updated in discrete time steps. Rules are updated with a function that depends on the current state of the cell itself and its neighbored cells. The rules can be deterministic, stochastic or empirical. There are several tools for the simulation of cellular automata. Chen et al. developed a rule-based stochastic cellular automata model to simulate the classical predator–prey Lotka–Volterra model [24].

Petri nets. Heiner et al. described a *Petri net*-based framework for modelling and analyzing biochemical pathways [28]. The framework unifies qualitative, stochastic and continuous paradigms. According to Heiner et al., each perspective adds its contribution to the understanding of the system. Heiner et al. modeled a *extracellular signal-regulated kinases (ERK) transduction pathway system* [28]. A pathway describes a sequence of interactions among molecules [51]. In the particular case of signal transduction pathway, there is a process, called *receptor*, which receives an external stimulus. The receptor triggers a chain of reaction which end on some internal result.

Heiner et al. focused on transient behavior analysis of the transduction system. The stochastic Petri net describes a system of stochastic reaction rate equations and the continuous Petri net is an structured description of ordinary differential equations. Note that arcs in Petri can be annotated with stoichiometric information. Qualitative Petri nets can be analyzed by a branching-time temporal logic. The stochastic Petri nets can be analyzed by a continuous stochastic logic. As usual, the continuous model replaces the discrete values of species with continuous values which represent the overall behavior via concentrations. This means that the concentrations of a particular species in such a model will have the same value at each point of time for repeated experiments. Continuous model can also be analyzed with a linear-time temporal logic.

Grid Systems. *Grid Systems* is a formalism to model population dynamics proposed by Barbuti et al. [3]. The formalism is inspired by concepts of P systems and spatiality dynamics of cellular automata. In *Grid Systems*, environmental events that change population behavior can be defined as rewrite rules. There is a simulation tool for *Grid Systems* developed in Java [3].

Grid Systems was used to model a population model of a species of mosquitoes, *Aedes albopictus*, that considers three types of external events: temperature change, rainfall, and desiccation [3]. The events change the behavior of the species directly or indirectly. Each individual in the population can move around in the ecosystem.

5. Conclusions

In this article, we presented different formal languages for individual-based modelling of ecosystems. Some are based on process calculi, some on P systems and others on cellular automata or Petri nets. All these existing formal languages are complementary to each other and should not be seen as competition.

We presented several stochastic formalisms, for instance, stochastic process calculi and stochastic P systems. An advantage of stochastic formalisms is that they have semantics in terms of continuous-time Markov chains. It is well-known that continuous-time Markov chains for large numbers of individuals can be approximated by ordinary differential equations (ODE) and there are several numeric methods that are computationally efficient to simulate ODEs. The disadvantage of these formalisms is that, sometimes, it is more intuitive to model systems with discrete time because stochastic rates are not known and, instead, we only know the probabilities that an individual will change from one state to another.

We also presented several probabilistic formalisms, for instance, probabilistic process calculi and probabilistic P systems. As we argued before, an advantage of this type of formalisms is that, sometimes, it is easier to model systems in discrete time by considering the probabilities that an individual will execute a certain action, instead of modelling stochastic rates. However, the semantics of these languages suffer from state-space explosion, since they have an exponential number of states. A solution for this problem is to represent the average behavior of the system using mean-field equations, as it was done for *WSCCs* and *S-PALPS*. Another disadvantage of these formalisms is that they usually consider both probabilistic and non-deterministic behavior. This semantics limits the properties that can be verified for these systems because non-deterministic behavior cannot be approximated by probabilities.

There is another interesting category of formalisms which are spatial extensions of probabilistic and stochastic process calculi or P systems. The semantics of these modelling languages is either stochastic or probabilistic, but they offer simpler ways to represent space – either continuously or discrete –, which facilitates the modelling of complex temporal properties of population systems.

Spatial, probabilistic and stochastic modelling languages have some drawbacks as modelling formalisms for ecosystems. Garavel argues that models based on process calculi are not widespread because there are many calculi and many variants for each calculus, being difficult to choose the most appropriate [32]. Garavel also argued that existing tools for process calculi are not user-friendly. We claim that these problems are also present in P systems and other formalisms to model population systems.

A research direction is how to develop user-friendly tools to simulate process calculi and P systems; for instance, to develop web applications or mobile applications that do not require extensive knowledge on Unix, LaTeX or programming to be used. This could help to make these formalisms spread world wide.

Another research direction is to analyze and simulate larger case studies, studying how different evolutions can arise, depending on different assumptions, such as spatial configurations, interaction rates and the role of the environment, high-level reasoning about them. Since spatial dynamics and temporal evolution of ecological systems are of key interest, an interesting further step will be to develop a complementary spatio-temporal logic, suitable for formally describing and verifying spatio-temporal properties of the different formalisms. A research on that direction is the three-valued spatio-temporal logic developed by Luisa Vissat et al. [42].

Finally, we argue that research in formal modelling languages for population systems in ecology has produced a vast corpus of deep, valuable results; however, these results are only known, understood and applied by a small fraction of computer scientists. As a consequence, their practical impact is not as strong as it could be, in spite of successful attempts at using process calculi, P systems and other approaches to model and verify properties of ecological systems. We believe, however, that these formal modelling languages still have an important role to play in the future, but more interdisciplinary work is needed in collaboration with ecologists, software developers and policy makers.

References

- [1] R. Barbuti, A. Bompadre, P. Bove, P. Milazzo, G. Pardini, Attributed probabilistic p systems and their application to the modelling of social interactions in primates, in: Proceedings of SEFM 2015, in: Lect. Notes Comput. Sci., vol. 9509, Springer, 2015, pp. 176–191.
- [2] R. Barbuti, P. Bove, A. Maggiolo-Schettini, P. Milazzo, G. Pardini, A computational formal model of the invasiveness of eastern species in European water frog populations, in: Proceedings of MOKMASD 2013, in: Lect. Notes Comput. Sci., vol. 8368, Springer, 2013, pp. 207–218.
- [3] R. Barbuti, A. Cerone, A. Maggiolo-Schettini, P. Milazzo, S. Setiawan, Modelling population dynamics using grid systems, in: Proceedings of SEFM 2012, in: Lect. Notes Comput. Sci., vol. 7991, Springer, 2014, pp. 172–189.
- [4] R. Barbuti, A. Maggiolo-Schettini, P. Milazzo, G. Pardini, Spatial calculus of looping sequences, Theor. Comput. Sci. 412 (43) (2011) 5976–6001.
- [5] A. Barco, S. Knight, F.D. Valencia, K-stores: a spatial and epistemic concurrent constraint interpreter, in: Proceedings of WFLP 2012, Informal Proceedings, 2012.
- [6] E. Bartocci, D.R. Cacciagnano, M.R.D. Berardini, E. Merelli, L. Tesei, Timed operational semantics and well-formedness of shape calculus, Sci. Ann. Comput. Sci. 20 (2010) 32–52.
- [7] E. Bartocci, F. Corradini, M.R.D. Berardini, E. Merelli, L. Tesei, Shape calculus. A spatial mobile calculus for 3D shapes, Sci. Ann. Comput. Sci. 20 (2010) 1–31.
- [8] T.A. Basuki, A. Cerone, R. Barbuti, A. Maggiolo-Schettini, P. Milazzo, E. Rossi, Modelling the dynamics of an Aedes albopictus population, in: Proceedings of AMCA-POP 2010, in: EPTCS, vol. 33, Open Publishing Association, 2010, pp. 18–36.
- [9] S. Benkirane, R. Norman, E. Scott, C. Shankland, Measles epidemics and pepa: an exploration of historic disease dynamics using process algebra, in: Proceedings of FM 2012, in: Lect. Notes Comput. Sci., vol. 7436, Springer, 2012, pp. 101–115.
- [10] D. Besozzi, P. Cazzaniga, D. Pescini, G. Mauri, Modelling metapopulations with stochastic membrane systems, Biosystems 91 (3) (2008) 499–514.
- [11] D. Besozzi, P. Cazzaniga, D. Pescini, G. Mauri, An analysis on the influence of network topologies on local and global dynamics of metapopulation systems, in: Proceedings of AMCA-POP 2010, in: EPTCS, vol. 33, Open Publishing Association, 2010, pp. 1–17.
- [12] L. Bioglio, C. Calcagno, M. Coppo, F. Damiani, E. Sciacca, S. Spinella, A. Troina, A spatial calculus of wrapped compartments, CoRR, arXiv:1108.3426, 2011.
- [13] L. Bortolussi, J. Hillston, D. Latella, M. Massink, Continuous approximation of collective system behaviour: a tutorial, Perform. Eval. 70 (5) (2013) 317–349.
- [14] L. Bortolussi, A. Policriti, Stochastic concurrent constraint programming and differential equations, Electron. Notes Theor. Comput. Sci. 190 (3) (2007) 27–42.
- [15] L. Bortolussi, A. Policriti, Modeling biological systems in stochastic concurrent constraint programming, Constraints 13 (1) (Jun 2008) 66–90.
- [16] L. Bortolussi, A. Policriti, Hybrid dynamics of stochastic pi-calculus, Math. Comput. Sci. 2 (3) (2009) 465–491.
- [17] M. Calder, S. Gilmore, J. Hillston, Modelling the influence of RKIP on the ERK signalling pathway using the stochastic process algebra PEPA, in: Transactions on Computational Systems Biology VII, in: Lect. Notes Comput. Sci., vol. 4230, Springer, 2006, pp. 1–23.
- [18] L. Cardelli, Brane calculi – interactions of biological membranes, in: Proceedings of CMSB 2004, in: Lect. Notes Comput. Sci., vol. 3082, Springer, 2005, pp. 257–278.
- [19] L. Cardelli, P. Gardner, Processes in space, in: Proceedings of GIE 2010, in: Lect. Notes Comput. Sci., vol. 6158, Springer, 2010, pp. 78–87.
- [20] L. Cardelli, G. Ghelli, A.D. Gordon, Types for the ambient calculus, Inf. Comput. 177 (2) (2002) 160–194.
- [21] L. Cardelli, R. Mardare, Stochastic Pi-Calculus Revisited, Technical report, Microsoft Research, Cambridge, UK, 2012.
- [22] M. Cardona, M.A. Colomer, A. Margalida, A. Palau, I. Pérez-Hurtado, M.J. Pérez-Jiménez, D. Sanuy, A computational modeling for real ecosystems based on P systems, Nat. Comput. 10 (1) (2011) 39–53.
- [23] P. Cazzaniga, D. Pescini, D. Besozzi, G. Mauri, Tau leaping stochastic simulation method in P systems, in: Proceedings of WMC 2006, in: Lect. Notes Comput. Sci., vol. 4361, Springer, 2006, pp. 298–313.
- [24] Q. Chen, F. Ye, W. Li, Cellular-automata-based ecological and ecohydraulics modelling, J. Hydroinform. 11 (3/4) (2009) 252–272.
- [25] F. Ciocchetta, J. Hillston, Bio-PEPA: a framework for the modelling and analysis of biological systems, Theor. Comput. Sci. 410 (33–34) (2009) 3065–3084.
- [26] F. Ciocchetta, J. Hillston, Bio-pepa for epidemiological models, Electron. Notes Theor. Comput. Sci. 261 (2010) 43–69.
- [27] M.A. Colomer, I. Pérez-Hurtado, M.J. Pérez-Jiménez, A. Riscos-Núñez, Comparing simulation algorithms for multienvironment probabilistic p systems over a standard virtual ecosystem, Nat. Comput. 11 (3) (2012) 369–379.
- [28] M. Curti, P. Degano, C. Priami, C. Baldari, Modelling biochemical pathways through enhanced π -calculus, Theor. Comput. Sci. 325 (1) (2004) 111–140.
- [29] R. De Nicola, D. Latella, M. Loreti, M. Massink, Rate-based transition systems for stochastic process calculi, in: Proceedings of ICALP 2009, in: Lect. Notes Comput. Sci., vol. 5556, Springer, 2009, pp. 435–446.
- [30] C. Feng, J. Hillston, Paloma: a process algebra for located markovian agents, in: Proceedings of QEST 2014, in: Lect. Notes Comput. Sci., vol. 8657, Springer, 2014, pp. 265–280.
- [31] V. Galpin, Modelling network performance with a spatial stochastic process algebra, in: Proceedings of AINA 2009, IEEE Computer Society, 2009, pp. 41–49.
- [32] H. Garavel, Reflections on the future of concurrency theory in general and process calculi in particular, Electron. Notes Theor. Comput. Sci. 209 (2008) 149–164.
- [33] S. Gilmore, J. Hillston, The PEPA workbench: a tool to support a process algebra-based approach to performance modelling, in: Proceedings of TOOLS 1994, in: Lect. Notes Comput. Sci., vol. 794, Springer, 1994, pp. 353–368.
- [34] O.M. Herescu, C. Palamidessi, Probabilistic asynchronous pi-calculus, in: Proceedings of FOSSACS 2000, in: Lect. Notes Comput. Sci., vol. 1784, Springer, 2000, pp. 146–160.
- [35] J. Hillston, M. Tribastone, S. Gilmore, Stochastic process algebras: from individuals to populations, Comput. J. 55 (7) (2012) 866–881.
- [36] C.A.R. Hoare, Communicating Sequential Processes, Prentice-Hall, 1985.
- [37] M. John, R. Ewald, A.M. Uhrmacher, A spatial extension to the pi-calculus, Electron. Notes Theor. Comput. Sci. 194 (3) (2008) 133–148.
- [38] M. John, C. Lhoussaine, J. Niehren, A.M. Uhrmacher, The attributed pi-calculus with priorities, in: Transactions on Computational Systems Biology XII, in: Lect. Notes Comput. Sci., vol. 5945, Springer, 2010, pp. 13–76.
- [39] D. Kouzapas, A. Philippou, A process calculus for dynamic networks, in: Proceedings of FMOODS/FORTE 2011, in: Lect. Notes Comput. Sci., vol. 6722, Springer, 2011, pp. 213–227.
- [40] M. Kwiatkowska, G. Norman, D. Parker, Using probabilistic model checking in systems biology, SIGMETRICS Perform. Eval. Rev. 35 (4) (2008) 14–21.
- [41] M. Kwiatkowska, I. Stark, The continuous π -calculus: a process algebra for biochemical modelling, in: Proceedings of CMSB 2008, in: Lect. Notes Comput. Sci., vol. 5307, Springer, 2008, pp. 103–122.
- [42] L. Luisa Vissat, J. Hillston, M. Loreti, L. Nenzi, Automatic verification of reliability requirements of spatio-temporal analysis using three-valued spatio-temporal logic, in: Proceedings of VALUETOOLS 2017, ACM, 2017, pp. 225–226.
- [43] L. Luisa Vissat, J. Hillston, G. Marion, M.J. Smith, MELA: modelling in ecology with location attributes, in: Proceedings of QAPL 2016, in: EPTCS, vol. 227, Open Publishing Association, 2016, pp. 82–97.

- [44] C. McCaig, R. Norman, C. Shankland, Process algebra models of population dynamics, in: Proceedings of AB 2008, in: Lect. Notes Comput. Sci., vol. 5147, Springer, 2008, pp. 139–155.
- [45] C. McCaig, R. Norman, C. Shankland, From individuals to populations: a mean field semantics for process algebra, Theor. Comput. Sci. 412 (17) (2011) 1557–1580.
- [46] M. Miculan, G. Bacci, Modal logics for brane calculus, in: C. Priami (Ed.), Proceedings of CMSB 2006, in: Lect. Notes Comput. Sci., vol. 4210, Springer, 2006, pp. 1–16.
- [47] R. Milner, A Calculus of Communicating Systems, Springer, 1980.
- [48] R. Milner, A Finite Delay Operator in Synchronous CCS, University of Edinburgh, Department of Computer Science, Edinburgh, UK, 1982.
- [49] R. Milner, Communicating and Mobile Systems – The Pi-Calculus, Cambridge University Press, 1999.
- [50] G. Norman, C. Palamidessi, D. Parker, P. Wu, Model checking probabilistic and stochastic extensions of the π -calculus, IEEE Trans. Softw. Eng. 35 (2) (2009) 209–223.
- [51] G. Pardini, Formal Modelling and Simulation of Biological Systems with Spatiality, PhD thesis, University of Pisa, 2011.
- [52] J. Pérez, C. Rueda, Non-determinism and probabilities in timed concurrent constraint programming, in: Proceedings of ICLP 2008, in: Lect. Notes Comput. Sci., vol. 5366, Springer, 2008, pp. 677–681.
- [53] D. Pescini, D. Besozzi, G. Mauri, C. Zandron, Dynamical probabilistic P-systems, J. Found. Comput. Sci. 17 (1) (2006) 183–204.
- [54] A. Philippou, M. Toro, Process ordering in a process calculus for spatially-explicit ecological models, in: Proceedings of MOKMASD 2013, in: Lect. Notes Comput. Sci., vol. 8368, Springer, 2013, pp. 345–361.
- [55] A. Philippou, M. Toro, M. Antonaki, Simulation and verification for a process calculus for spatially-explicit ecological models, Sci. Ann. Comput. Sci. 23 (1) (2013) 119–167.
- [56] A. Phillips, L. Cardelli, A correct abstract machine for the stochastic pi-calculus, in: Proceedings of Bioconcur 2004, in: Electron. Notes Theor. Comput. Sci., 2004.
- [57] J.C. Prentice, G. Marion, P.C.L. White, R.S. Davidson, M.R. Hutchings, Demographic processes drive increases in wildlife disease following population reduction, PLoS ONE 9 (5) (05 2014) 1–11.
- [58] C. Priami, P. Quaglia, Beta binders for biological interactions, in: Proceedings of CMSB 2004, in: Lect. Notes Comput. Sci., vol. 3082, Springer, 2005, pp. 20–33.
- [59] G. Păun, Computing with membranes (P systems): an introduction, in: Current Trends in Theoretical Computer Science, World Scientific, 2001, pp. 845–866.
- [60] G. Păun, Membrane Computing: An Introduction, Springer-Verlag, 2002.
- [61] A. Regev, E.M. Panina, W. Silverman, L. Cardelli, E. Shapiro, BioAmbients: an abstraction for biological compartments, Theor. Comput. Sci. 325 (1) (Sept. 2004) 141–167.
- [62] F.J. Romero-Campero, M. Gheorghe, L. Bianco, D. Pescini, M. Pérez-Jiménez, R. Ceterchi, Towards probabilistic model checking on P systems using PRISM, in: WMC 2006, in: Lect. Notes Comput. Sci., vol. 4361, Springer, 2006, pp. 477–495.
- [63] I. Scagnetto, M. Miculan, Ambient calculus and its logic in the calculus of inductive constructions, Electron. Notes Theor. Comput. Sci. 70 (2) (2002) 76–95.
- [64] A. Spicher, O. Michel, M. Cieslak, J.-L. Giavitto, P. Prusinkiewicz, Stochastic P systems and the simulation of biochemical processes with dynamic compartments, Biosystems 91 (3) (2008) 458–472.
- [65] D.J.T. Sumpter, G.B. Blanchard, D.S. Broomhear, Ants and agents: a process algebra approach to modelling ant colony behaviour, Bull. Math. Biol. 63 (2001) 951–980.
- [66] J. Sun, S. Song, Y. Liu, Model checking hierarchical probabilistic systems, in: Proceedings of ICFEM 2010, in: Lect. Notes Comput. Sci., vol. 6447, Springer, 2010, pp. 388–403.
- [67] C. Tofts, Processes with probabilities, priority and time, Form. Asp. Comput. 6 (5) (1994) 536–564.
- [68] M. Toro, Probabilistic extension to the concurrent constraint factor oracle model for music improvisation, Intel. Artif. 57 (19) (2016) 37–73.
- [69] M. Toro, M. Desainte-Catherine, C. Rueda, Formal semantics for interactive music scores: a framework to design, specify properties and execute interactive scenarios, J. Math. Music 8 (1) (2014) 93–112.
- [70] M. Toro, A. Philippou, S. Arboleda, M. Puerta, C.M. Vélez S., Mean-field semantics for a process calculus for spatially-explicit ecological models, in: DCM 2015, in: EPTCS, vol. 204, Open Publishing Association, 2016, pp. 79–94.
- [71] M. Toro, A. Philippou, C. Kassara, S. Sfenthourakis, Synchronous parallel composition in a process calculus for ecological models, in: Proceedings of ICTAC 2014, in: Lect. Notes Comput. Sci., vol. 8687, 2014, pp. 424–441.
- [72] M. Toro, C. Rueda, C. Agón, G. Assayag, Ntcrt: a concurrent constraint framework for soft real-time music interaction, J. Theor. Appl. Inf. Technol. 82 (1) (2015) 184–193.
- [73] M. Tribastone, S. Gilmore, J. Hillston, Scalable differential analysis of process algebra models, IEEE Trans. Softw. Eng. 38 (1) (2012) 205–219.
- [74] H.M. Yang, C.P. Ferreira, Assessing the effects of vector control on dengue transmission, Appl. Math. Comput. 198 (2008) 401–413.