A process calculus for spatially-explicit ecological models

Xenia Efthymiou Anna Philippou
Department of Computer Science
University of Cyprus

cs06ep@cs.ucy.ac.cy annap@cs.ucy.ac.cy

In this paper we propose PALPS, a Process Algebra with Locations for Population Systems. PALPS allows us to produce spatially-explicit, individual-based models and to reason about their behavior. Our calculus has two levels: at the first level we may define the behavior of an individual of a population while, at the second level, we may specify a system as the collection of individuals of various species located in space, moving through their life cycle while changing their location, if they so wish, and interacting with each other in various ways such as preying on each other. We describe the syntax and the semantics of PALPS and we illustrate its applicability via simple examples.

1 Introduction

During the last years we have witnessed an increasing trend towards the use of formal frameworks for reasoning about biological as well as ecological systems including process algebras [14, 15, 11], cellular automata [6] and *P*-systems [4]. Process algebras, first proposed in [13, 8] to aid the understanding and reasoning about concurrent systems, have proved to provide a number of features that make them amenable towards capturing biological processes. In particular, process algebras are especially suited towards the so-called individual-based approach of modeling populations as they enable one to describe the evolution of each individual of the population as a process and, subsequently, to compose a set of individuals (as well as their environment) into a complete ecological system. Features such as time, probability and stochastic behavior, which have been extensively studied within the context of process algebras, can be exploited to provide more accurate models, while associated analysis tools can be used to analyze and predict their behavior.

In this work, our aim is to introduce a process-algebraic framework equipped with the notion of a *location* to enable spatially-explicit modeling of ecological systems. In particular, we propose a domain-specific process algebra which associates individuals with information about their position and thus allows to explore location-dependent behavior of a population system. There exists a variety of existing work which introduces location behavior into formal frameworks. Amongst them, we mention [2, 9, 1] which introduce the concept of a location into frameworks developed for reasoning about biological processes. Relevant proposals that introduce locations in process algebras for reasoning about ad hoc networks can be found in [12, 7, 10] while more recently [3] introduces a π-calculus equipped with the notion of space. The novelty of our proposal is that it associates location information with population-system specific behavior such as reproduction and preying. In the next section we present our process calculus and in the final section we conclude with remarks on future work.

2 The Process Calculus

In our calculus, PALPS (Process Algebra with Locations for Population Systems), we consider a system as a set of individuals operating in space, each possessing a species and a location identifier. Movement in the calculus is modeled via a specialized action whose effect is to change the location of an individual, with the restriction that the originating and the destination locations are neighboring locations. The notion of neighborhood is implemented via a relation **Nb** where $(\ell, \ell') \in \mathbf{Nb}$ exactly when locations ℓ and ℓ' are neighbors.

2.1 The Syntax

We continue to formalize the syntax of PALPS. We begin by describing the basic entities of the calculus. We assume a set of channels **Ch**, ranged over by a, b, as well as a set of locations **Loc** ranged over by ℓ, ℓ'. Furthermore, we assume a set of special labels **S** corresponding to the species under consideration, ranged over by $\mathbf{s}$, $\mathbf{s}'$. To model preying, we assume the existence of a relation $\mathbf{Prey} \subseteq \mathbf{S} \times \mathbf{S}$, where $(\mathbf{s}, \mathbf{s}') \in \mathbf{Prey}$ if individuals of species $\mathbf{s}$ prey on individuals of species $\mathbf{s}'$.

Our calculus also employs a set of logical expressions ranged over by e. One of our main aims being to facilitate reasoning about spatial-dependent behavior, these conditions are intended to capture environmental (location-relevant) situations which may affect the behavior of individuals. In the present form of PALPS we limit conditions about the environment to observations on the number of individuals of the same or another species co-existing within the same location. Thus, we consider expressions e, to be built as follows:

$$e \quad ::= \quad true \mid \neg e \mid e_1 \wedge e_2 \mid (\mathbf{s}@\ell) \bowtie c \mid @\ell \bowtie c$$

where c is a natural number and $\bowtie \in \{=, \leq, \geq\}$, the intention being that $(\mathbf{s}@\ell) \bowtie c$ is satisfied in a system if the number of individuals of species $\mathbf{s}$ at location ℓ are equal to / less than / greater than c. We also write $@\ell \bowtie c$ to denote that the total number of individuals of all species at location ℓ are $\bowtie c$. Note that we allow ℓ to also take the special value myloc. This label refers to the actual location of the individual which is checking the containing condition, no matter which this location might be, and it is instantiated to this location when the individual is placed within a context and a location (see rule (Loc) in Table 2). We then write $S \models e$ for a population system S and an expression e, exactly when S satisfies e. The relation $\models$ is defined by induction on e in Table 4.

The syntax of PALPS consists of three levels: (1) the individual level (ranged over by P), (2) the species level (ranged over by R) and (3) the system level (ranged over by S). Their syntax is defined via the following BNF's

$$\begin{array}{rcl} P & ::= & \mathbf{0} \mid \eta.P \mid P_1 + P_2 \mid \ \mathsf{cond}\,(e_1 \rhd P_1, \ldots, e_n \rhd P_n) \mid C \\ R & ::= & !a.P \\ S & ::= & \mathbf{0} \ \mid \ P{:}[\![\mathbf{s}, \ell]\!] \ \mid \ R{:}[\![\mathbf{s}]\!] \ \mid \ S_1 \,|\, S_2 \ \mid \ S \backslash L \ \mid \ [S] \end{array}$$

where $a \in \mathbf{Ch}$, $L \subseteq \mathbf{Ch}$, C ranges over a set of process constants $\mathscr{C}$, each with an associated definition of the form $C \stackrel{\mathrm{def}}{=} P$, where the node P may contain occurrences of C, as well as other constants, and

$$\eta ::= a \mid \overline{a} \mid move \mid prey \mid \surd .$$

Beginning with the *individual* level P, the $\mathbf{0}$ process represents the inactive individual. $\eta.P$ describes the individual who first engages in activity η and then behaves as P. Activity η can be an (input) action on a channel a, written simply as a, a complementary (output) action on a channel a, written as $\overline{a}$, a movement action, *move*, a preying action, *prey*, or a time-passing action, $\surd$. Actions of the form a, and $\overline{a}$, $a \in \mathbf{Ch}$, are used to model arbitrary activities performed by an individual e.g. eating, observing the environment as well as reproduction. A $\surd$ action measures a tick on a global clock and is used to separate the phases/rounds of an individual's behavior. Essentially, given a system, the intention is that in any given time unit all individuals perform their available actions, possibly synchronizing as necessary, until they synchronize on their next $\surd$ action. $P_1 + P_2$ represents the nondeterministic choice between P_1 and P_2. The conditional process $\mathsf{cond}\ (e_1 \rhd P_1, \ldots, e_n \rhd P_n)$ presents the conditional choice between a set of processes: it behaves as P_i, where i is the smallest integer for which e_i evaluates to $\mathtt{true}$. Finally, process constants provide a mechanism for including recursion in the calculus.

Moving on to the *species* level, we note that during their life cycle, individuals may produce offsprings. To capture the creation of new individuals, we employ the special *species* processes R. R, defined as $!a.P$, is a replicated process which may continuously receive input through channel a. This results in the creation of a new individual P. Such inputs will be provided by individuals in the phase of reproduction.

Finally, population systems are built on the basis of located individuals, $P{:}[\![\mathbf{s},\ell]\!]$, where $\mathbf{s}$ and ℓ are the species and the location of the individual, and species $C{:}[\![\mathbf{s}]\!]$, where $\mathbf{s}$ is the name of the species. Furthermore, $S\backslash L$ models the restriction of the use of channels in set L within S and $[S]$ is the closure operator. This operator is applied at the highest level of a population system and its semantic significance is that it allows us to select the valid behavior of the system based on the conditions that the system satisfies.

As an example, consider the model described in [2] where a set of individual live on an $n \times n$ lattice of resource sites and go through phases of reproduction and dispersal. Specifically, the studied model considers a population where individuals disperse in space while competing for a location site during their reproduction phase. They produce an offspring only if they have exclusive use of a location. After reproduction the offsprings disperse and continue indefinitely with the same behavior. In PALPS, we may model the described species $\mathbf{s}$ as $!rep.P$, where

$$\begin{aligned} P &\stackrel{\mathrm{def}}{=} move.\surd.\mathsf{cond}\ (\mathbf{s}@\mathsf{myloc} = 1 \rhd P_1;\ \mathsf{true} \rhd \surd.P) \\ P_1 &\stackrel{\mathrm{def}}{=} \overline{rep}.\surd.P_1 + \overline{rep}.\overline{rep}.\surd.P_1 \end{aligned}$$

We point out that the conditional construct allows us to determine the exclusive use of a location by an individual. The special label myloc is used to illustrate that the location of interest is the actual location of an individual once the individual is placed in a context within a system definition. Furthermore, note that P_1 models the nondeterministic production of one or two offsprings of the species. During the dispersal phase, an individual moves to a neighboring location which is chosen nondeterministically, as prescribed in the semantics of the next section. Then a system containing of two individuals at a location ℓ and one in location ℓ' can be modeled as

$$System \stackrel{\mathrm{def}}{=} [P{:}[\![\ell,\mathbf{s}]\!] | P{:}[\![\ell,\mathbf{s}]\!] | P{:}[\![\ell',\mathbf{s}]\!] | (!rep.P){:}[\![\mathbf{s}]\!]].$$

To model a competing species $\mathbf{s}'$ which preys on $\mathbf{s}$, we may define the process $!rep'.Q$, where

$$\begin{aligned} Q &\stackrel{\mathrm{def}}{=} prey.\surd.Q_1 + \surd.Q_2 \\ Q_1 &\stackrel{\mathrm{def}}{=} \overline{rep'}.\surd.Q \\ Q_2 &\stackrel{\mathrm{def}}{=} prey.\surd.Q_1 + \surd.die.\mathbf{0} \end{aligned}$$

This species looks for a prey. If it succeeds it produces an offspring. If it fails for two consecutive time units its dies.

The notion of food at a location may also be modeled in PALPS. A channel *eat* is employed to model eating and, for example, a food source at location ℓ of amount n which replenishes every t time units can be described as $Food_{n,t}{:}[\![f,\ell]\!]$, where

$$Food_{i,j} \stackrel{\text{def}}{=} \begin{cases} eat.Food_{i-1,j} + \surd.Food_{i,j-1} & \text{if } i>0, j>0 \\ \surd.Food_{i,j-1} & \text{if } i=0 \\ Food_{n,t} & \text{if } j=0 \end{cases}$$

2.2 The Semantics

The semantics of PALPS is defined in terms of a structural operational semantics, which is given in Tables 1-3. The rules of Table 1 describe the behavior of individuals in isolation, the rules in Table 2 the behavior of systems and the rule of Table 3 describes the behavior of closed systems. A transition of P has the form $P \overset{e,\eta}{\twoheadrightarrow} P'$, specifying that P can perform action η under condition e and evolve into P'.

Table 1: **Transition rules for individuals**

(Nil)	$\mathbf{0} \overset{true,\surd}{\twoheadrightarrow} \mathbf{0}$	(Act)	$\eta.P \overset{true,\eta}{\twoheadrightarrow} P$
(Sum)	$\dfrac{P_i \overset{e,\alpha}{\twoheadrightarrow} P_i', i\in\{1,2\}}{P_1+P_2 \overset{e,\alpha}{\twoheadrightarrow} P_i'}$	(Const)	$\dfrac{P \overset{e,\alpha}{\twoheadrightarrow} P'}{C \overset{e,\alpha}{\twoheadrightarrow} P'} \quad C \stackrel{\text{def}}{=} P$
(Cond)	$\dfrac{P_i \overset{e,\alpha}{\twoheadrightarrow} P_i', e' = e_i \wedge (\bigwedge_{j<i} \neg e_i)}{\mathsf{cond}\ (e_1 \rhd P_1, \ldots, e_n \rhd P_n) \overset{e\wedge e',\alpha}{\twoheadrightarrow} P_i'}$		

Axiom (Nil) specifies that the inactive process may allow time to pass while Axiom (Act) states that $\eta.P$ can always execute action η and evolve to P. Rules (Sum) and (Const) express the semantics of nondeterministic choice and process constants in the expected way, where (Cond) stipulates that a conditional process may perform an (conditional) action of continuation P_i assuming that e_i evaluates to True and all e_j, $j<i$ are false.

Moving on to the higher level of the semantics, a transition of S has the form $S \xrightarrow{e,\alpha} S'$, signifying that S can perform action α under condition e and evolve into S'. Actions α can have one of the following forms:

- $a@\ell$ and $\overline{a}@\ell$ denote the execution of actions a and $\overline{a}$ respectively at location ℓ.
- $prey_{\mathbf{s}}@\ell$ denotes the execution of a prey action at location ℓ by an individual belonging to the species $\mathbf{s}$.
- τ denotes the internal action. This may arise when two complementary actions take place at the same location or when a move or a prey action take place. We are not interested in the precise location of internal actions, thus, this information is omitted.
- $\surd$ denotes the time passing action.

Note that in Table 2 we write θ to range over all η actions with the exception of the specialized actions *move* and *prey*, which are treated separately, and β to range over all α actions with the exception of action $prey_{\mathbf{s}}@\ell$.

Table 2: **Transition rules for systems**

$$(\mathsf{Loc})\quad \frac{P \xrightarrow{e,\theta}\!\!\!\!\!\rightarrow P'}{P{:}[\![\mathbf{s},\ell]\!] \xrightarrow{e@\ell,\theta@\ell} P'{:}[\![\mathbf{s},\ell]\!]} \qquad (\mathsf{Move})\quad \frac{P \xrightarrow{e,move}\!\!\!\!\!\rightarrow P',\ (\ell,\ell')\in \mathbf{Nb}}{P{:}[\![\mathbf{s},\ell]\!] \xrightarrow{e,\tau} P'{:}[\![\mathbf{s},\ell']\!]}$$

$$(\mathsf{Par1})\quad \frac{S_1 \xrightarrow{e,\beta} S_1'}{S_1|S_2 \xrightarrow{e,\beta} S_1'|S_2} \qquad (\mathsf{Par2})\quad \frac{S_1 \xrightarrow{e_1,a@\ell} S_1',\ S_2 \xrightarrow{e_2,\overline{a}@\ell} S_2'}{S_1|S_2 \xrightarrow{e_1\wedge e_2,\tau} S_1'|S_2'}$$

$$(\mathsf{RepA})\quad \frac{\ell\in \mathbf{Loc}}{(!a.P){:}[\![\mathbf{s}]\!] \xrightarrow{true,a@\ell} P{:}[\![\mathbf{s},\ell]\!]|(!a.P){:}[\![\mathbf{s}]\!]} \qquad (\mathsf{RepT})\quad (!a.P){:}[\![\mathbf{s}]\!] \xrightarrow{true,\surd} (!a.P){:}[\![\mathbf{s}]\!]$$

$$(\mathsf{PreyS})\quad \frac{S_1 \xrightarrow{e,prey_{\mathbf{s}}@\ell} S_1',\ \ \mathbf{s}'\in Preys(\mathbf{s}),\ \ S\equiv (S_1\mid S_2\mid P{:}[\![\mathbf{s}',\ell]\!])\backslash L}{S \xrightarrow{e,\tau} (S_1'\mid S_2)\backslash L} \qquad (\mathsf{PreyI})\quad \frac{P \xrightarrow{e,prey}\!\!\!\!\!\rightarrow P'}{P{:}[\![\mathbf{s},\ell]\!] \xrightarrow{e,prey_{\mathbf{s}}@\ell} P'{:}[\![\mathbf{s},\ell']\!]}$$

$$(\mathsf{Hide})\quad \frac{S \xrightarrow{e,\alpha} S',\ \alpha\notin\{a@\ell,\overline{a}@\ell|a\in L\}}{S\backslash L \xrightarrow{e,\alpha} S'\backslash L} \qquad (\mathsf{Time})\quad \frac{S_1 \xrightarrow{e_1,\surd} S_1',\ S_2 \xrightarrow{e_2,\surd} S_2'}{S_1|S_2 \xrightarrow{e_1\wedge e_2,\surd} S_1'|S_2'}$$

To begin with, rule (Loc) embeds location information to actions of a located process. Note that we define $e@\ell$ by $e[\ell/\mathsf{myloc}]$, that is, the condition with all instances of myloc substituted by ℓ. Next, rule (Move) specifies that a located process may nondeterministically move to any neighboring location. Rules (Par1) and (Par2) stipulate the semantics of the parallel composition construct (their symmetric versions are omitted). Rule (RepA) defined the semantics of the replication construct. Here we may observe how the generator of new individuals may create a new located individual of a species while itself remaining in the environment for further use. Note that $(!a.P){:}[\![\mathbf{s}]\!]$ can communicate with individuals at all locations and the newly-instantiated individual acquires the location of its parent. Furthermore, a species may liberally allow time ticks (axiom (RepT)). Moving on to rules (PreyI) and (PreyS) we may see how a preying individual may kill an individual of a species on which it preys. Note that this rules is defined on top of a structural equivalence relation $\equiv$. The axioms of interest to us with respect to this relation are the following where $\mathsf{ch}(S)$ refers to the set of all channels in S.

$$\begin{array}{ll} S\equiv S\backslash\emptyset & S_1|S_2\equiv S_2|S_1 \\ (S\backslash L_1)\backslash L_2\equiv S\backslash(L_1\cup L_2) & (S_1|S_2)|S_3\equiv S_1|(S_2|S_3) \\ S\equiv S[a/b]\text{ , if } a\notin \mathsf{ch}(S) & S_1\backslash\{a\}|S_2\equiv (S_1|S_2)\backslash\{a\}\text{, if } a\notin\mathsf{ch}(S_2) \end{array}$$

Essentially, this relation allow us to rewrite a system while isolating a prey and a predator and, consequently, extinguish the prey according to the rule.

Rule (Hide) implements restriction of the set of channels in L and, finally, (Time) imposes a synchronous nature to the time-passing action $\surd$.

The transition relation we have just defined is a conditional relation: transitions are decorated by conditions that determine when they can occur. Naturally, we are interested in distilling the behavior of a system for conditions that evaluate to true. To achieve this, we have introduced the closure construct which should be applied at the top level of systems to signal their completion. Given such closed systems we may now evaluate the various conditions of the actions of transition system $\longrightarrow$ and select only those transitions whose condition evaluates to true. The new transition relation is symbolized as $\mapsto$ where we write $CS \overset{\alpha}{\mapsto} CS'$ whenever the closed system CS can perform action α and evolve to CS'. This relation is defined by the rule (Close) in Table 3.

Table 3: **Transition rule for closed systems**

$$\text{(Close)} \qquad \frac{S \xrightarrow{e,\alpha} S', S \models e}{[S] \overset{\alpha}{\mapsto} [S']}$$

The satisfaction relation $\models$ is defined inductively on the structure of systems as shown in Table 4.

Table 4: **The satisfaction relation**

$S \models \mathsf{true}$
$S \models \neg e$ if and only if $\neg(S \models e)$
$S \models e_1 \wedge e_2$ if and only if $S \models e_1 \wedge S \models e_2$
$S \models \mathbf{s}@\ell \bowtie c$ if and only if $\mathsf{num}(S,\ell,\mathbf{s}) \bowtie c$
$S \models @\ell \bowtie c$ if and only if $\mathsf{num}'(S,\ell) \bowtie c$

The auxiliary functions $\mathsf{num}(S,\ell,\mathbf{s})$ and $\mathsf{num}'(S,\ell)$ compute the number of individuals at location ℓ in system $\mathbf{S}$ of a specific species $\mathbf{s}$ ($\mathsf{num}(\ldots)$) or for all species ($\mathsf{num}'(\ldots)$). Their definition is straightforward. Below we present that of $\mathsf{num}(\ldots)$.

$$\begin{array}{rcl}
\mathsf{num}(\mathbf{0},\ell,\mathbf{s}) & = & 0 \\
\mathsf{num}(P{:}[\![\mathbf{s},\ell']\!],\ell,\mathbf{s}) & = & 1, \text{ if } \ell = \ell' \\
\mathsf{num}(P{:}[\![\mathbf{s},\ell']\!],\ell,\mathbf{s}) & = & 0, \text{ if } \ell \neq \ell' \\
\mathsf{num}(R{:}[\![\mathbf{s}]\!],\ell,\mathbf{s}) & = & 0 \\
\mathsf{num}(S_1 \,|\, S_2,\ell,\mathbf{s}) & = & \mathsf{num}(S_1,\ell,\mathbf{s}) + \mathsf{num}(S_2,\ell,\mathbf{s}) \\
\mathsf{num}(S \backslash L,\ell,\mathbf{s}) & = & \mathsf{num}(S,\ell,\mathbf{s}) \\
\mathsf{num}([S],\ell,\mathbf{s}) & = & \mathsf{num}(S,\ell,\mathbf{s})
\end{array}$$

3 Concluding remarks

This paper reports on work in progress towards the development of a process calculus for the spatially-explicit and individual-based modeling of ecological systems. In future work we intend to extend our study by developing the theory of the calculus and introducing probabilistic behavior. Most importantly, we plan to implement a tool to accompany our language for performing simulations and possibly analysis of modeled systems. In related work, we have in fact implemented a prototype tool for a variant of the calculus containing probabilistic choice, locations, movement and reproduction [5]. As our experiments have shown, and as one would expect, the notion of location increases the complexity of evaluating systems. Thus, our future work will also concentrate on providing optimizations for system analysis.

References

[1] R. Barbuti, A. Maggiolo-Schettini, P. Milazzo & G. Pardini (2009): *Spatial Calculus of Looping Sequences*. Electronic Notes in Theoretical Computer Science 229(1), pp. 21–39.

[2] A. Brännström & D. J. T. Sumpter (2005): *Coupled map lattice approximations for spatially explicit individual-based models of ecology*. Bulletin of Mathematical Biology 67(4), pp. 663–682.

[3] L. Cardelli & P. Gardner (2010): *Processes in space*. In: Proceedings of CiE 2010, LNCS 6158, Springer, pp. 78–87.

[4] M. Cardona, M. Colomer, A. Margalida, I. Pérez-Hurtado, M. J. Pérez-Jiménez & D. Sanuy (2010): *A P System Based Model of an Ecosystem of the Scavenger Birds*. In: Proceedings of WMC 2009, LNCS 5957, Springer, pp. 182–195.

[5] X. Efthymiou (2010): *A Process Algebraic Framework and Tool for Reasoning about Spacially Explicit Population Systems*. Undergraduate thesis, University of Cyprus.

[6] S. C. Fu & G. Milne (2004): *A Flexible Automata Model for Disease Simulation*. In: Proceedings of ACRI 2004, LNCS 3305, Springer, pp. 642–649.

[7] J. Ch. Godskesen (2009): *A Calculus for Mobile Ad-hoc Networks with Static Location Binding*. Electronic Notes in Theoretical Computer Science 242(1), pp. 161–183.

[8] C. A. R. Hoare (1985): *Communicating Sequential Processes*. Prentice-Hall.

[9] M. John, R. Ewalda & A. M. Uhrmacher (2008): *A Spatial Extension to the π-Calculus*. Electronic Notes in Theoretical Computer Science 194, pp. 133–148.

[10] D. Kouzapas & A. Philippou (2010): *A Process Calculus for Systems with Dynamic Topology*. Technical Report TR-10-01, University of Cyprus.

[11] C. McCaig, R. Norman & C. Shankland (2008): *Process Algebra Models of Population Dynamics*. In: Proceedings of AB 2008, LNCS 5147, Springer, pp. 139–155.

[12] M. Merro (2009): *An Observational Theory for Mobile Ad Hoc Networks*. Information and Computation 207(2), pp. 194–208.

[13] R. Milner (1980): *A Calculus of Communicating Systems*. Springer.

[14] C. Tofts (1993): *Using process algebra to describe social insect behaviour*. Transactions of the Society for Computer Simulation 9, pp. 227–283.

[15] C. Tofts (1994): *Processes with probabilities, priority and time*. Formal Aspects of Computing 6, pp. 536–564.