The Dynamics and Stability of Probabilistic Population Processes [⋆]

Ioannis Chatzigiannakis[1] and Paul G. Spirakis[2,3]

[1] Department of Computer, Control and Management Engineering, Sapienza University of Rome, Italy
[2] Computer Science Department, University of Liverpool, UK
[3] Computer Engineering and Informatics Department, Patras University, Greece
E-mails: ichatz@dis.uniroma1.it, p.spirakis@liverpool.ac.uk

Abstract. We study here the dynamics and stability of Probabilistic Population Processes, via the differential equations approach. We provide a quite general model following the work of Kurtz [20] for aproximating discrete processes with continuous differential equations. We show that it includes the model of Angluin et. al. [1], in the case of very large populations. We require that the long-term behavior of the family of increasingly large discrete processes is a good aproximation to the long-term behavior of the continuous process, i.e., we exclude population protocols that are extremely unstable such as parity-dependent decision processes. For the general model we give a sufficient condition for stability that can be checked in polynomial time. We also study two interesting subcases: (a) protocols whose specifications (in our terms) are configuration independent. We show that they are always stable and that their eventual subpopulation percentages are actually a Markov Chain stationary distribution. (b) protocols that have dynamics resembling virus spread. We show that their dynamics are actually similar to the well-known Replicator Dynamics of Evolutionary Games. We also provide a sufficient condition for stability in this case.

1 Introduction

In the near future, it is reasonable to expect that new types of systems will appear, designed or emerged, of massive scale, expansive and permeating their environment, of very heterogeneous nature, and operating in a constantly changing networked environment. Such systems are expected to operate even beyond the complete understanding and control of their designers, developers, and users. Although they will be perpetually adapting to a constantly changing environment, they will have to meet their clearly-defined objectives and provide guarantees about certain aspects of their own behavior.

We expect that most such systems will have the form of a very large society of networked artefacts. Each such artefact will be unimpressive: small, with limited sensing, signal processing, and communication capabilities, and usually of limited energy. Yet by cooperation, they will be organized in large societies to accomplish tasks that are difficult or beyond the capabilities of todays conventional centralized systems. These systems or societies should have particular ways to achieve an appropriate level of organization and integration. This organization should be achieved seamlessly and with appropriate levels of flexibility, in order to be able to achieve their global goals and objectives.

Angluin et al. [1, 2] introduced the notion of a computation by a population protocol to model such distributed systems in which individual agents are extremely limited and can be represented as finite state machines. In their model, finite-state, and complex behavior of the system as a whole emerges from the rules governing pairwise interaction of the agents. The computation is carried out by a collection of agents, each of which receives a piece of the input. These agents move around and information can be exchanged between two agents whenever they come into contact with each

[⋆] A previous version of some aspects of this work has appeared as a brief announcement in DISC 2008 [9] other. The goal is to ensure that every agent can eventually output the value that is to be computed (assuming a fairness condition on the sequence of interactions that occur).

In [1] they also proposed a natural probabilistic variation of the standard population protocol model, in which finite-state agents interact in pairs under the control of an adversary scheduler. In this variant, interactions that occurs between pairs of agents are chosen uniformly at random (i.e., by employing a random scheduler). We call the protocols of [1] by the term "Probabilistic Population Processes" (PPP). In [3] they presented fast algorithms for performing computations in this variation and showed how to use the notion of a leader in order to efficiently compute semilinear predicates and in order to simulate efficiently LOGSPACE Turing Machines. [12] studied the acquisition and propagation of knowledge in the probabilistic model of random interactions between all paris in a population (conjugating automata). A particular form of probabilistic population dynamics that is based on "baptising" the other member of the interaction was recently studied in [11]. The topic of population protocols has been studied recently towards establishing a broader understanding of the effects of local memory [22, 8], distrinct identiers [14], failures [4] and existence of leader [6].

A notable set of recent works has studied (distributed) computation in *worst-case* dynamic networks in which the topology may change arbitrarily from round to round (see e.g. [19, 23]). Another interesting direction assumes random network dynamicity and the interest is on determining "good" properties of the dynamic network that hold with high probability and on designing protocols for distributed tasks [10, 5]. For introductory texts cf. [7, 21, 26].

In this work we characterize the dynamics of population protocols by examining the rate of growth of the states of the agents as the protocol evolves. We imagine here a continuoum of agents. By the law of large numbers, one can model the undelaying aggregate stochastic process as a deterministic flow system. Our main proposal here is to exploit the powerful tools of continuous nonlinear dynamics in order to examine questions (such as stability) of such protocols. The use of differential equations to model the dynamics of distributed interactions has been briefly used in the past for task allocation in robot networks [13].

Such an approach was first suggested by the seminal work of Kurtz [20]. That approach approximates the behavior of a system of discrete dynamics with a system of differential equations in the limit. This also relates to Wormald's Lemma [28], taking into careful consideration the timing of the conversion of the discrete to a continuous analog. Here is a brief description of Wormald's Lemma: Given a stochastic process in which tokens of type 1, 2, 3, etc. interact with a probability that is a continuous function of their concentrations $x_1 overn$, $\frac{x_2}{n}$, etc. (where xi counts the number of tokens of type i), resulting in an increase or decrease of each xi by some constant determined by the particular interaction that occurs, then in the limit as we increase n (where n is the size of the population) while rescaling time as t/n we obtain a continuous process defined in terms of differential equations where the derivative of the x vector with respect to time is given by the sum of the various increments multiplied by their probabilities. Wormald's Lemma says that for any fixed time t/n, the distance between the discrete concentrations xi/n and the corresponding component of the solution to the differential equation is o(1) with high probability.

We first provide a very general model for population protocol continuous dynamics. This model (Switching Population Processes – SPP) includes the probabilistic population protocols (PPP) of [1] as a special case, when the population is infinite and the time is continuous.

We show a sufficient condition for stability of SPP that can be checked in *polynomial time*. We also examine two subclasses of SPP:

- The *Markovian Population Processes* (MAP). In these protocols, their *specifications are configuration independent*. In this very practical case, we show that MAP are *always stable* and their *unique* population mix at stability is exacly the steady-state distribution of a *Markov Chain*.
- The *Linear Viral Processes* (LVP). They are probabilistic protocols motivated by the "random pairing" of [1]. However, agents review their current state at a higher rate when they have weak "immunity". We view this as a general model for the dynamics of viruses spread in the population. We show that LVP are equivalent to the well-known "Replicator Dynamics" of Evolutionary Game Theory. We also give a sufficient condition for stability of LVP, based on potentials.

2 The General Model (Switching Probabilistic Processes – SPP)

The network is modeled as a complete graph G where vertices represent nodes and edges represent communication links between nodes. We use the letter n to denote $|V|$, the number of nodes in the network. Each node is capable of executing an "agent" (or process) which consists of the following components:

- K, a finite set of states. We use the letter k to denote $|K|$.
- X, a nonempty subset of K, known as the inital states or start states.

We consider a large population of n agents. Let $q \in K$ be a state of the agent and let n_q the number of agents that are on the given state p. Then the total population size is $n = \sum_{i=1}^{k} n_i$. The proportion of agents that are at state q is $x_q = \frac{n_q}{n}$. We call x_q the *density* of q. In the sequel $q = q_i$, where $i \in \{1, 2, \ldots, k\}$.

A state assignment of a system is defined to be an assignment of a state to each agent in the system. A *configuration* C is a map from the population to states, giving the current state of every agent. The population state density then, at time t, can be described via a vector $\boldsymbol{x}(t) = (x_1(t), \ldots, x_k(t))$. Here $x_i(t) = \frac{n_i}{n}$, $i = 1 \ldots k$.

In the sequel we assume that $n \to \infty$. We are interested, thus, in the evolution of $\boldsymbol{x}(t)$ as time goes on. We use a different model (compared to [1]) for describing a protocol P. We imagine that all agents in the population are infinitely lived and that they interact forever. Each agent sticks to some state in K for some time interval, and now and then *reviews* her state. This depends on $\boldsymbol{x}(t)$ and may result to a change of state of the agent. Based on this concept, a *switching population protocol* consists of the following two basic elements (specifications):

1. A specification of the *time rate* at which agents in the population review their state. This rate may depend on the current, "local", performance of the agent's state and also on the configuration $\boldsymbol{x}(t)$.
2. A specification of the *switching probabilities* of a reviewing agent. The probability that an agent, currently in state q_i at a review time, will *switch* to state q_j is in general a function $p_{ij}\left(\boldsymbol{x}(t)\right)$, where $p_i\left(\boldsymbol{x}\right) = \left(p_{i1}\left(\boldsymbol{x}\right), \ldots, p_{ik}\left(\boldsymbol{x}\right)\right)$ is the resulting distribution over the set K of states in the protocol.

In a large, finite, population n, we assume that the review times of an agent are the "birth times" of a Poisson process of rate $\lambda_i\left(\boldsymbol{x}\right)$. At each such time, the agent i selects a new state according to $p_i\left(\boldsymbol{x}\right)$. We assume that all such Poisson processes are independent. Then, the aggregate of review times in the sub-population of agents in state q_i is itself a Poisson process of birth rate $x_i \lambda_i(\boldsymbol{x})$. As in the probabilistic model of [1] we assume that state switches are independent random variables accross agents. Then, the rate of the (aggregate) Poisson process of switches from state q_i to state q_j in the whole population is just $x_i(t)\lambda_i(\boldsymbol{x}(t))\, p_{ij}(\boldsymbol{x}(t))$.

When $n \to \infty$, we can model the aggregate stochastic processes as deterministic flows (see, e.g., [24, 25, 28]). The outflow from state q_i is $\sum_{j \neq i} x_j \lambda_j(\boldsymbol{x})\, p_{ij}(\boldsymbol{x})$. Then, the rate of change of $x_i(t)$ (i.e. $\frac{dx_i(t)}{dt}$ or $\dot{x}_i(t)$) is just

$$\dot{x}_i \;=\; \sum_{j \in K} x_j p_{ji}(\boldsymbol{x})\, \lambda_j(\boldsymbol{x}) \;-\; \lambda_i(\boldsymbol{x})\, x_i \tag{1}$$

for $i = 1, \ldots, k$.

We assume here that both $\lambda_i(\boldsymbol{x})$ and $p_{ij}(\boldsymbol{x})$ are Lipschitz continuous functions in an open domain $\varSigma$ containing the simplex $\varDelta$ where

$$\varDelta \;=\; \left\{ (x_i, \ldots, x_k) : \sum_{i=1}^{K} x_i = 1\,, \qquad x_i \geq 0\;,\; \forall i \right\}$$

By the theorem of Picard-Linderlöf (see, e.g., [16] for a proof), Eq. 1 has a *unique* solution for any initial state $\boldsymbol{x}(0)$ in $\varDelta$ and such a solution trajectory $\boldsymbol{x}(t)$ is *continuous* and never leaves $\varDelta$.

2.1 SPP includes the probabilistic population protocols

We now show that our model of Switching Probabilistic Processes (SPP) is more general than the model of [1] in the sense that it can be used to define the Probabilistic Population Processes (PPP). We do this by showing the following:

Theorem 1. The continuous time dynamics of PPP (when $n \to \infty$) are a special case of the dynamics of SPP.

Proof. According to [1], the discrete-time dynamics of a Probabilistic Population Protocol (PPP) are given by a finite set of rules, R of the form

$$(p, q) \mapsto (p', q')$$

where $p, q, p', q' \in K$ ($K = \{q_1, \ldots, q_k\}$) together with a set A of n agents and an (irreflexive) relation $E \subseteq A \times A$.

Intruitively, a $(u, v) \in E$ means that u, v are able to interact. [1] assumes further that E consists of all ordered pairs of distinct elements from A.

A *population configuration* in [1] is a mapping $C : A \mapsto K$ (K is the set of states). Let C and C' be population configurations, and u, v be two distinct agents. [1] says that C can go to C' in one discrete step (denoted $C \overset{e}{\mapsto} C'$) via an *encounter* $e = (u, v)$ if

$$(C(u), C(v)) \mapsto (C'(u), C'(v))$$

is a rule in R. This means that the state $C(u)$ of u switches to $C'(u)$ and also $C(v)$ switches to $C'(v)$.

The execution of the system is defined to be a sequence $C_0, C_1, C_2, \ldots$ of configurations (where C_0 is the initial configuration) such that for each i, $C_i \mapsto C_{i+1}$. An execution is fair if for any C_i

and C_j, such that $C_i \mapsto C_j$ and C_i occurs infinitely often in the execution, C_j also occurs infinitely often in the execution.

In the probabilistic version of the above, [1] further states that e (the ordered pair to interact) is chosen at random, independently and uniformly from all ordered pairs corresponding to edges e in $A \times A$ ([1] calls it the model of Conjugating Automata, inspired also by [12]).

Let us now assume that $n \to \infty$ and let $x_i = \lim_{n\to\infty} \frac{n_i}{n}$ be the population fraction at state $q_i \in K$ at a particular configuration C, at time t. Consider the rule ρ in R

$$(q_r, q_m) \mapsto (q_i, q_j)$$

Without loss of generality we assume in the sequel that $r \neq m$ and $i \neq j$ in such rules ρ in R. By the uniformity and randomness, the probability that such an e, that follows from rule ρ, is selected (as the encounter), is just $x_r(t)x_m(t)$. Let A_i be the set of all (r, m) that are the left part of a rule ρ:

$$\begin{array}{rl} & (q_r, q_m) \mapsto (q_i, q_j) \\ \text{or} & (q_r, q_m) \mapsto (q_j, q_i) \end{array}$$

Let B_i be the set of (r, m) that are the left part of a rule ρ':

$$(q_r, q_m) \mapsto (q_{r'}, q_{m'})$$

with $r = i$ or $m = i$. Without loss of generality let $r = i$ in ρ'. By considering a small interval Δt and taking limits as $\Delta t \to 0$, due to fairness we get $\forall i$:

$$\dot{x_i} = \sum_{(r,m)\in A_i} x_r(t)x_m(t) - x_i(t) \sum_{(i,m)\in B_i} x_m(t) \tag{2}$$

The above set of equations describe the continuous dynamics of PPP.

Now, consider our SPP dynamics and Eq. 1. Set $\lambda_i(\boldsymbol{x}) = \sum x_m(t)$, with m ranging over all rules

$$(q_r, q_m) \mapsto (q_{r'}, q_{m'})$$

with $r = i$, and all rules

$$(q_m, q_r) \mapsto (q_{r'}, q_{m'})$$

with $r = i$ (i.e., over all rules in B_i).

Also, set $p_{mi} = p_{ri}$ = 0, if r, m do not belong in any tuple of A_i.

Finally set

$$p_{ri} = \frac{1}{\lambda_r} \sum_{m\in C(r,i)} x_m(t)$$

where $C(r, i)$ is the set of indices m in the second argument of the left part of rules in A_i (i.e. $(q_r, q_m) \mapsto (q_{r'}, q_{m'})$ with $r' = i$ or $m' = i$).

Then our system of Eq. 1 (the SPP dynamics) becomes the system of Eq. 3 (the PPP dynamics). Thus the PPP dynamics are a special case of the SPP dynamics in the continuous time setting. □

Here is an example of the reduction described above. Let the rules R in PPP be

$$\begin{aligned}
(q_1, q_2) &\mapsto (q_3, q_2) \\
(q_3, q_1) &\mapsto (q_1, q_2) \\
(q_2, q_3) &\mapsto (q_2, q_1)
\end{aligned}$$

This gives the continuous PPP dynamics:

$$\begin{aligned}
\dot{x_1} &= x_1 x_3 \,+\, x_2 x_3 \,-\, x_1 \left(x_2 + x_3\right) \\
\dot{x_2} &= x_1 x_3 \,+\, x_1 x_2 \,+\, x_2 x_3 \,-\, x_2 \left(x_1 + x_3\right) \\
\dot{x_3} &= x_1 x_2 \,-\, x_3 \left(x_1 + x_2\right)
\end{aligned}$$

We then set

$$\begin{aligned}
\lambda_1 &= x_2 \,+\, x_3 \\
\lambda_2 &= x_1 \,+\, x_3 \\
\lambda_3 &= x_1 \,+\, x_2
\end{aligned}$$

and

$$\begin{array}{lll}
p_{21} = \frac{x_3}{x_1+x_3} & p_{11} = \frac{x_3}{x_2+x_3} & p_{31} = 0 \\
p_{12} = \frac{x_3}{x_2+x_3} & p_{22} = \frac{x_1}{x_1+x_3} & p_{32} = \frac{x_2}{x_1+x_2} \\
p_{13} = \frac{x_2}{x_2+x_3} & p_{23} = p_{33} = 0 &
\end{array}$$

and this results in our SPP dynamics, namely:

$$\begin{aligned}
\dot{x_1} &= x_1 \lambda_1 p_{11} \,+\, x_2 \lambda_2 p_{21} \,+\, x_3 \lambda_3 p_{31} \,-\, x_1 \lambda_1 \\
\dot{x_2} &= x_1 \lambda_1 p_{12} \,+\, x_2 \lambda_2 p_{22} \,+\, x_3 \lambda_3 p_{32} \,-\, x_2 \lambda_2 \\
\dot{x_3} &= x_1 \lambda_1 p_{13} \,+\, x_2 \lambda_2 p_{23} \,+\, x_3 \lambda_3 p_{33} \,-\, x_3 \lambda_3
\end{aligned}$$

3 Stability of nonlinear dynamic systems: a sufficient condition for decidability.

Let us consider a dynamic system

$$\dot{x_i} \;=\; f_i\left(\boldsymbol{x}\right) \,, \qquad i = 1, \ldots, k$$

that is, in fact, more general than Eq. 1.

Definition 1 (Fixed Points). Let $\boldsymbol{x}^*$ be a solution of the system $\{f_i\left(\boldsymbol{x}^*\right) = 0,\, i = 1, \ldots, k\}$ which we call a *fixed point* of the system.

By making a Taylor expansion around $\boldsymbol{x}^*$ we obtain a linear approximation to the dynamics:

$$\dot{x_i} \;=\; \sum \left(x_j - x_j^*\right) \frac{df_i}{dx_j} \left(\boldsymbol{x}^*\right)$$

Setting $\xi_i = x_i - x_i^*$ we get

$$\dot{\xi}_i \;=\; \sum \xi_j \frac{df_i}{dx_j}\left(\boldsymbol{x}^*\right)$$

which is a Linear System with a fixed point at the origin, i.e., $\dot{\xi} \;=\; L\xi$ where the matrix L has *constant* components $L_{ij} \;=\; \frac{df_i}{dx_j}\left(\boldsymbol{x}^*\right)$. L is called the Jacobian Matrix. Then, by the theorem of [15] we have

Corollary 1. If the fixed point $\boldsymbol{x}^*$ is *hyperbolic* (i.e., all eigenvalues of L^* have a non-zero real part) then the topology of the dynamics of the nonlinear system around $\boldsymbol{x}^*$ is the same as the topology of a $\boldsymbol{x}^*$ in the Linear system.

In fact, let each eigenvalue of L be $\phi = \mathsf{a} + i\omega$.

Corollary 2. Let $\mathsf{a} \neq 0$, $\forall\phi$ eigenvalues of L. Then

(a) If $\mathsf{a} < 0$, $\forall\phi$ then $\boldsymbol{x}(t)$ approaches the fixed point $\boldsymbol{x}^*$ as $t \to \infty$.

(b) If there exists a ϕ with $\mathsf{a} > 0$ then $\boldsymbol{x}(t)$ *diverges* from the fixed point $\boldsymbol{x}^*$ along the direction of the corresponding eigenvector. That is, the fixed point $\boldsymbol{x}^*$ is unstable.

Thus we get our main result of the system:

Theorem 2. If all fixed points $\boldsymbol{x}^*$ of our population dynamics of Eq. 1 are hyperbolic, then we *can decide stability* of the population protocol, around x^*, in *polynomial time* in the description of the protocol.

Corollary 3. If all fixed points of PPP are hyperbolic, then the stability of PPP can be decided in polynomial time.

4 Switching Population Processes with specifications independent of the configuration

We now consider the special case of Eq. 1 where $\lambda_i\left(\boldsymbol{x}\right) = \lambda_i \; \forall i$ and where $p_{ij}\left(\boldsymbol{x}\right) = p_{ij}$ (specifications independent of the configuration $\boldsymbol{x}(t)$). Then the basic system of Eq. 1 of the dynamics of the population becomes:

$$\dot{x_i} \;=\; \sum_{j\in K} x_j \lambda_j p_{ji} \;-\; \lambda_i x_i \qquad\qquad i = 1\ldots k \tag{3}$$

We call such protocols by the term "Markovian Population Processes" (MAP).

Let $q_{ij} = \lambda_i p_{ij}$ for all i, j, when $i \neq j$ and when $j = i$ let $q_{ii} = \lambda_i(p_{ii} - 1)$. Then Eq. 3 in fact becomes

$$\frac{dx_i(t)}{dt} \;=\; q_{ii}x_i(t) \;+\; \sum_{j\neq i} q_{ki}x_k(t) \tag{4}$$

Note that $\sum_{i\in K} x_i(t) = 1$. But this is, in fact, the basic equation of the limiting-state probabilities of a Markov Chain of k states with q_{ij} being the (continuous time) rates of change (see, e.g., [18], pp. 53–55).

When all λ_{ij}, $i \neq j$ are non zero then the Markov Chain of Eq. 4 is irreducible and homogeneous. Then the limits $\lim_{t \to \infty} x_i(t)$ always exist and are independent of the initial state. The limiting distribution is given *uniquely* as the solution of the following equations:

$$q_{jj}x_j + \sum_{k \neq j} q_{kj}x_k = 0$$

So, we get our second major result:

Theorem 3 (Markovian Population Processes – MAP). Let the specifications $\{\lambda_j, p_{ij}\}$ independent of the configuration $\boldsymbol{x}(t)$. Let also $\lambda_j p_{ij} \neq 0$, $\forall i, j$ where $i \neq j$. Then the Population Protocol is *stable*. It always has a limiting *unique* configuration $\{x_i \;\; i = 1 \ldots k\}$ independent of the initial configuration $\boldsymbol{x}(0)$, which is exactly the *steady-state distribution* of an *ergodic, homogeneous Markov Chain of k states*.

5 A special case of Random pairing population protocols (Linear Viral Processes – LVP)

Now, let us assume that all reviewing agents adopt the state of "the first man they meet in the street". This is clearly the case when the reviewing agent draws a pairing agent at random from the population (according to the uniform probability distribution across agents) and adopts the state of the so sampled agent. This is similar to the case of the protocols of [1] where the rules are $(q_i, q_k) \mapsto (q_m, q_r)$ with $r, m \in \{i, j\}$. Formally then

$$p_{ij}\left(\boldsymbol{x}\right) = x_j \qquad \forall i, j \in K, \, \forall x(t)$$

Now Eq. 4 becomes

$$\dot{x_i} = \sum_{j \in K} x_j x_i \lambda_j(x) - \lambda_i(x) x_i$$

i.e.

$$\dot{x_i} = \left(\sum_{j \in K} x_j \lambda_j(x) - \lambda_i(x) \right) \cdot x_i \tag{5}$$

We now propose a "linear" model in order to capture the immunity that an agent has against other agents in the population. We postulate that agents immunity depend on their states. So all agents at state state experience the same immunity. One can imagine immunity to be a measure of the degree of protection of agents when they interact. So, when an agent in state q_i interacts with an agent in state q_j we measure the immunity of the (q_i, q_j) pair by an integer a_{ij} and we require here that $\mathsf{a}_{ij} = \mathsf{a}_{ji}$ (we assume symmetric interactions). It is then natural to assume that agents in state q_i will wish to review their state more often when their immunity is low. In particular we assume here that any agent in state q_i has a review rate $\lambda_i\left(\boldsymbol{x}\right)$ that is *linearly decreasing* in the average immunity of the agent in state q_i. This is the simplest possible model. The formal definitions follow:

Definition 2 (Immunity of a state). *Let $A = \{\mathsf{a}_{ij}\}$ be a symmetric matrix of integers. The immunity of an agent in state q_i is $t_i\left(\boldsymbol{x}\right) = \mathsf{a}_{i1}x_1 + \ldots + \mathsf{a}_{ik}x_k$.*

Definition 3 (Average immunity of a population protocol, in a particular configuration). *Let A be a symmetric matrix of integers. The average immunity of the population, in configuration $\{x_i\}$, is: $t\left(\boldsymbol{x}\right) = \sum_{i \in K} x_i t_i\left(\boldsymbol{x}\right)$.*

Definition 4 (Linear Viral Processes – LVP). *The Linear Viral Processes are switching population protocols with review rates of agents*

$$\lambda_i\left(\boldsymbol{x}\right) = \gamma - \delta t_i\left(\boldsymbol{x}\right)$$

where $\gamma, \delta \in \Re$, $\delta > 0$ and also $\gamma/\delta \geq t_i\left(\boldsymbol{x}\right)$, $\forall \boldsymbol{x} + \mathit{\Delta}$, $\forall i$.

Now Eq. 5 becomes

$$\dot{x_i} \;=\; \delta\left(t_i\left(\boldsymbol{x}\right) - t\left(\boldsymbol{x}\right)\right) x_i \tag{6}$$

Note, now, that this equation is a constant rescaling of the popular "replicator dynamics" of Evolutionary Game Theory (see, e.g., [27]).

Definition 5. *The general Lotka-Volterra equation for k types of a population is of the form*

$$\dot{x_i} \;=\; x_i\left(r_i + \sum_{j=1}^{k} a_{ij} x_j\right) \qquad i = 1 \ldots k$$

where r_i, a_{ij} are constant.

By the equivalence of the Replicator Dynamics with the Lotka-Volterra systems we then get:

Theorem 4. The dynamics of the linear viral protocols are *equivalent* to the Lotka-Volterra dynamics.

We can then give an alternative sufficient condition for the (asymptotic) stability of the Linear Viral Processes.

Theorem 5. Let x^* be a fixed point of Eq. 6, i.e., $t_i\left(\boldsymbol{x}\right) = t\left(\boldsymbol{x}\right)$ is satisfied for $\boldsymbol{x} = \boldsymbol{x}^*$. If $\sum_{i=1}^{k} x_i^* t_i\left(\boldsymbol{x}\right) > t\left(\boldsymbol{x}\right)$ for any $\boldsymbol{x}$ in a region around $\boldsymbol{x}^*$, then $\boldsymbol{x}^*$ is asymptotically stable.

In order to prove our theorem, we first consider the relative entropy of $\boldsymbol{x}$ and $\boldsymbol{x}^*$ as

$$E(x) = -\sum_{i=1}^{k} x_i^* \ln\left(\frac{x_i}{x_i^*}\right) \tag{7}$$

Clearly $E(x^*) = 0$. Then we need to prove the following claim:

Claim. $E(x) \geq E(x^*), \forall \boldsymbol{x}$

Proof. From Jensen's inequality it folds:

$$\exp\left(f(x)\right) \;\geq\; f(\exp x)$$

where $\exp()$ is the expectation, x a random variable and f a convex function. Thus Eq. 7 becomes

$$E(x) \geq -\ln\left(\sum_{i=1}^{k} x_i^* \frac{x_i}{x_i^*}\right) \geq -\ln\left(\sum_{i=1}^{k} x_i\right) = -\ln 1 = 0$$

□

Proof. Based on Claim 5 we can prove Theorem 5 as follows:

$$\begin{aligned}\frac{dE\left(\boldsymbol{x}(t)\right)}{dt} &= \sum_{i=1}^{k} \frac{dE}{dx_i}\dot{x}_i \\ &= -\sum_{i=1}^{k} \frac{x_i^*}{x_i}\dot{x}_i \\ &= -\sum_{i=1}^{k} \delta\left(t_i\left(\boldsymbol{x}\right) - t\left(\boldsymbol{x}\right)\right) x_i^* \qquad \text{(due to Eq. 6)} \\ &= -\delta\left[\sum_{i=1}^{k} \boldsymbol{x}^*\left(t_i(x) - t\left(\boldsymbol{x}\right)\right)\right] \\ &< 0 \qquad \text{by assumption}\end{aligned}$$

Thus, in a region around $\boldsymbol{x}^*$, $\frac{dE}{dt} < 0$. Then E is a (strict) Lyapounov function (see, e.g., [17], pp. 18–19) and thus $\boldsymbol{x}^*$ is stable asymptotically. □

6 Conclusions

The *population protocol* model of Angluin et. al. [1] consists of a (large) population of finite-state *agents* that interact in pairs. Each interaction updates the state of both participants according to a transition function based on the pair of the participants' previous states. A natural probabilistic model, proposed in [1], assumes each interaction to occur between a pair of agents chosen uniformly at random. We call the protocols of [1] by the term "Probabilistic Population Processes" (PPP). [12] studied the acquisition and propagation of knowledge in the probabilistic model of random interactions between all paris in a population (conjugating automata). Curiously, the differential equation approach for such protocols was not proposed till now.

We imagine here a continuoum of agents. By the law of large numbers, one can model the underlying aggregate stochastic process as a deterministic flow system. Our main proposal here is to exploit the powerful tools of continuous nonlinear dynamics in order to examine questions (such as stability) of such protocols.

We have extended the class of [1] by defining a general model of "Switching Population Processes" (SPP). We then examined stability for this general model and two important subclasses. Our main point is that one can study stability and population dynamics of protocols, via nonlinear differential equations that describe quite accurately the (discrete) population protocol dynamics when the population is very large. The "differential equations" approach was indicated in the past for the analysis of evolution of algorithms with Random Inputs, by [25, 24, 28]. Our approach provides a sufficient condition for stability of PPP of [1] that can be checked in polynomial time. It also gives a more general way to *specify* population protocols, that reveals interesting classes. A potential problem with this approach is that the long-term behavior of the continuous process may not be a good approximation to the long-term behavior of the family of increasingly large discrete processes it is supposed to describe in some cases. For example, it is not hard to construct a population process that converges with high probability to a configuration in which all tokens say EVEN if the number of 1 bits in the original population is even and ODD otherwise (this is a consequence of the LOGSPACE computation results of Angluin et al. [1] reference in the paper). No continuous limit can distinguish between these odd and even initial configurations, since we can approach any given limit concentration arbitrarily using only odd or even initial configurations. This is not a problem for Wormald's Lemma [28] (the time needed to distinguish between odd and even grows faster than n, so any for any fixed time t/n, the behavior of the discrete process doesn't depend much on parity yet), and it's not a problem for the earlier work of Kurtz [20] (which uses similar time scaling), but it should be a problem here, since the goal of the paper seems to be to describe the behavior of very large probabilistic population protocols. In the cases studied in this paper, this is not a problem, because the paper implicitly makes the same scaling assumption as this previous work, which makes everything interesting happen at a time pushed off into the infinite future. This limits the applicability of the results to finite processes. However such highly unstable protocols have limited usage and can be analysed with other techniques.

References

1. Dana Angluin, James Aspnes, Zoë Diamadi, Michael J. Fischer, and René Peralta. Computation in networks of passively mobile finite-state sensors. In *23rd Annual ACM Symposium on Principles of Distributed Computing (PODC)*, pages 290–299, New York, NY, USA, 2004. ACM.
2. Dana Angluin, James Aspnes, Zoë Diamadi, Michael J. Fischer, and René Peralta. Computation in networks of passively mobile finite-state sensors. *Distributed Computing*, 18(4):235–253, 2006.
3. Dana Angluin, James Aspnes, and David Eisenstat. Fast computation by population protocols with a leader. In *20th International Symposium on Distributed Computing (DISC)*, volume 4167 of *Lecture Notes in Computer Science*, pages 61–75. Springer, 2006.
4. Dana Angluin, James Aspnes, and David Eisenstat. A simple population protocol for fast robust approximate majority. In *21st International Symposium on Distributed Computing (DISC)*, volume 4731 of *Lecture Notes in Computer Science*, pages 20–32. Springer, 2007.
5. Chen Avin, Michal Koucký, and Zvi Lotker. How to explore a fast-changing world (cover time of a simple random walk on evolving graphs). In Luca Aceto, Ivan Damgård, Leslie Ann Goldberg, Magnús M. Halldórsson, Anna Ingólfsdóttir, and Igor Walukiewicz, editors, *Automata, Languages and Programming, 35th International Colloquium, ICALP 2008, Reykjavik, Iceland, July 7-11, 2008, Proceedings, Part I: Tack A: Algorithms, Automata, Complexity, and Games*, volume 5125 of *Lecture Notes in Computer Science*, pages 121–132. Springer, 2008.
6. Amanda Belleville, David Doty, and David Soloveichik. Hardness of Computing and Approximating Predicates and Functions with Leaderless Population Protocols. In Ioannis Chatzigiannakis, Piotr Indyk, Fabian Kuhn, and Anca Muscholl, editors, *44th International Colloquium on Automata, Languages, and Programming (ICALP 2017)*, volume 80 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 141:1–141:14, Dagstuhl, Germany, 2017. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.
7. Arnaud Casteigts, Paola Flocchini, Walter Quattrociocchi, and Nicola Santoro. Time-varying graphs and dynamic networks. *IJPEDS*, 27(5):387–408, 2012.
8. Ioannis Chatzigiannakis, Othon Michail, Stavros Nikolaou, Andreas Pavlogiannis, and Paul G. Spirakis. Passively mobile communicating machines that use restricted space. *Theor. Comput. Sci.*, 412(46):6469–6483, 2011.
9. Ioannis Chatzigiannakis and Paul G. Spirakis. The dynamics of probabilistic population protocols. In Gadi Taubenfeld, editor, *Distributed Computing, 22nd International Symposium, DISC 2008, Arcachon, France, September 22-24, 2008. Proceedings*, volume 5218 of *Lecture Notes in Computer Science*, pages 498–499. Springer, 2008.
10. Andrea E. F. Clementi, Claudio Macci, Angelo Monti, Francesco Pasquale, and Riccardo Silvestri. Flooding time in edge-markovian dynamic graphs. In Rida A. Bazzi and Boaz Patt-Shamir, editors, *Proceedings of the Twenty-Seventh Annual ACM Symposium on Principles of Distributed Computing, PODC 2008, Toronto, Canada, August 18-21, 2008*, pages 213–222. ACM, 2008.
11. Jurek Czyzowicz, Leszek Gasieniec, Adrian Kosowski, Evangelos Kranakis, Paul G. Spirakis, and Przemyslaw Uznanski. On convergence and threshold properties of discrete lotka-volterra population protocols. In Magnús M. Halldórsson, Kazuo Iwama, Naoki Kobayashi, and Bettina Speckmann, editors, *Automata, Languages, and Programming - 42nd International Colloquium, ICALP 2015, Kyoto, Japan, July 6-10, 2015, Proceedings, Part I*, volume 9134 of *Lecture Notes in Computer Science*, pages 393–405. Springer, 2015.
12. Zoë Diamadi and Michael J. Fischer. A simple game for the study of trust in distributed systems. *Wuhan University Journal of Natural Sciences*, 6(1-2):72–82, March 2001. Also appears as Yale Technical Report TR-1207, January 2001, `ftp://ftp.cs.yale.edu/pub/TR/tr1207.ps`.

13. Aram Galstyan and Kristina Lerman. Analysis of a stochastic model of adaptive task allocation in robots. In Sven Brueckner, Giovanna Di Marzo Serugendo, Anthony Karageorgos, and Radhika Nagpal, editors, *Engineering Self-Organising Systems, Methodologies and Applications [revised versions of papers presented at the Engineering Selforganising Applications (ESOA 2004) workshop, held during the Autonomous Agents and Multi-agent Systems conference (AAMAS 2004) in New York in July 2004, and selected invited papers]*, volume 3464 of *Lecture Notes in Computer Science*, pages 167–179. Springer, 2004.
14. Rachid Guerraoui and Eric Ruppert. Names trump malice: Tiny mobile agents can tolerate byzantine failures. In *36th Internatilonal Collogquium on Automata, Languages and Programming (ICALP 2009)*, volume 5556 of *Lecture Notes in Computer Science*, pages 484–495. Springer, 2009.
15. Philip Hartman. A lemma in the theory of structural stability of differential equations. *American Mathematical Society*, 11(4):610–620, 1960.
16. Morris Hirsch and Stephen Smale. *Differential Equations, Dynamical Systems and Linear Algebra*. Academic Press, 1974.
17. Josef Hofbauer and Karl Sigmund. *Evolutionary Games and Population Dynamics*. Cambridge University Press, 1998.
18. L. Kleinrock. *Queueing Systems, Theory*, volume I. John Wiley & Sons, 1975.
19. Fabian Kuhn, Nancy A. Lynch, and Rotem Oshman. Distributed computation in dynamic networks. In *Proceedings of the 42nd ACM Symposium on Theory of Computing, STOC 2010, Cambridge, Massachusetts, USA, 5-8 June 2010*, pages 513–522, 2010.
20. Thomas G Kurtz. Approximation of population processes, 1981.
21. Othon Michail, Ioannis Chatzigiannakis, and Paul Spirakis. *New Models for Population Protocols*. Synthesis Lectures on Distributed Computing Theory, N. Lynch Edt. Morgan & Claypool, 2011.
22. Othon Michail, Ioannis Chatzigiannakis, and Paul G. Spirakis. Mediated population protocols. *Theor. Comput. Sci.*, 412(22):2434–2450, 2011.
23. Othon Michail, Ioannis Chatzigiannakis, and Paul G. Spirakis. Causality, influence, and computation in possibly disconnected synchronous dynamic networks. In Roberto Baldoni, Paola Flocchini, and Binoy Ravindran, editors, *Principles of Distributed Systems, 16th International Conference, OPODIS 2012, Rome, Italy, December 18-20, 2012. Proceedings*, volume 7702 of *Lecture Notes in Computer Science*, pages 269–283. Springer, 2012.
24. Michael Mitzenmacher. Analyses of load stealing models based on families of differential equations. *Theory Comput. Syst.*, 34(1):77–98, 2001.
25. Michael Mitzenmacher and Eli Upfal. *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, 2005.
26. Christian Scheideler. Models and techniques for communication in dynamic networks. In Helmut Alt and Afonso Ferreira, editors, *STACS 2002, 19th Annual Symposium on Theoretical Aspects of Computer Science, Antibes - Juan les Pins, France, March 14-16, 2002, Proceedings*, volume 2285 of *Lecture Notes in Computer Science*, pages 27–49. Springer, 2002.
27. Jörgen W. Weibull. *Evolutionary Game Theory*. MIT Press, 1997.
28. Nicholas C. Wormald. Differential equations for random processes and random graphs. *Annals of Applied Probability*, 5:1217–1235, 1995.