# On the computational capabilities of several models*

Olivier Bournez and Emmanuel Hainry

LORIA/INRIA, 615 Rue du Jardin Botanique
54602 Villers-Lès-Nancy, FRANCE
{Olivier.Bournez,Emmanuel.Hainry}@loria.fr

**Abstract** We review some results about the computational power of several computational models. Considered models have in common to be related to continuous dynamical systems.

## 1 Dynamical Systems and Polynomial Cauchy Problems

A polynomial Cauchy problem is a Cauchy problem of type

$$\begin{cases} \mathbf{x}' &= p(\mathbf{x}, t) \\ \mathbf{x}(0) &= \mathbf{x}_0 \end{cases}$$

where $p(\mathbf{x}, t)$ is a vector of polynomials, and $\mathbf{x}_0$ is some initial condition.

The class of functions that are solution of a polynomial Cauchy problem turns out to be a very robust class [14]. It contains almost all natural mathematical functions. It is closed under addition, subtraction, multiplication, division, composition, differentiation, and compositional inverse [14].

Actually, every continuous time dynamical system $\mathbf{x}' = f(\mathbf{x}, t)$ where each component of $f$ is defined as a composition of functions in the class and polynomials can be shown equivalent to a (possibly higher dimensional) polynomial Cauchy problem [14]. This implies that almost all continuous time dynamical systems considered in books like [16], or [21] can be turned in the form of (possibly higher dimensional) polynomial Cauchy problems.

For example, consider the dynamic of a pendulum $x'' + p^2 \sin(x) = 0$. Because of the sin function, this is not directly a polynomial ordinary differential equation. However, define $y = x'$, $z = \sin(x)$, $u = \cos(x)$. A simple computation of derivatives show that we must have

$$\begin{cases} x' = y \\ y' = -p^2 z \\ z' = yu \\ u' = -yz \end{cases},$$

which is a polynomial ordinary differential equation.

* This work has been partially supported by French Ministry of Research through ANR Project SOGEA.

This class of dynamical systems becomes even more interesting if one realizes that it captures all what can be computed by some models of continuous time machines, such as the General Purpose Analog Computer (GPAC) of Shannon [25].

## 2 The GPAC

The GPACs was introduced in 1941 by Shannon [25] as a mathematical model of an analog device: the Differential Analyzer [10]. The Differential Analyzer was used from the 1930s to the early 60s to solve numerical problems. For example, differential equations were used to solve ballistics problems. These devices were first built with mechanical components and later evolved to electronic versions.

A GPAC may be seen as a circuit built of interconnected black boxes, whose behavior is given by Figure 1, where inputs are functions of an independent variable called the *time* (in an electronic Differential Analyzer, inputs usually correspond to electronic voltages). These black boxes add or multiply two inputs, generate a constant, or solve a particular kind of Initial Value Problem defined with an ordinary differential equation.

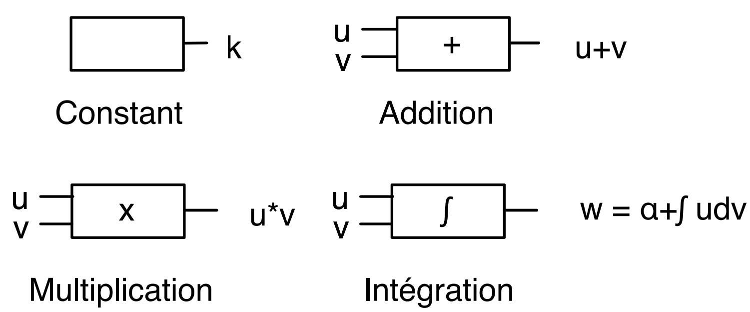


**Figure 1.** The basic units of a GPAC (the output $w$ of an integration operator satisfies $w'(t) = u(t)v'(t)$, $w(t_0) = \alpha$ for some initial condition $\alpha$).

The model was further refined in [24,19,13,15]. For the more robust class of GPACs defined in [15], the following property holds:

**Theorem 1 (GPAC Generated Functions [15]).** *A scalar function $f : \mathbb{R} \to \mathbb{R}$ is generated by a GPAC iff it is a component of the (necessarily unique) solution of a polynomial Cauchy problem. A function $f : \mathbb{R} \to \mathbb{R}^k$ is generated by a GPAC iff all of its components are.*

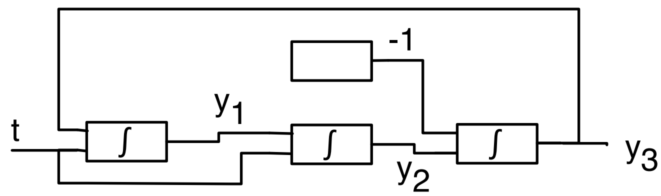


**Figure 2.** Generating cos and sin by a GPAC. In form of a system of equations, we have $y_1' = y_3$, $y_2' = y_1$, $y_3' = -y_1$. It follows that $y_1 = \cos$, $y_2 = \sin$, $y_3 = -\sin$, if $y_1(0) = 1$, $y_2(0) = y_3(0) = 0$.

From previous closure properties, GPAC can be considered as a faithful model of (today's) analog electronics [**?**]. Figure 3 shows how to realize an integration with an ideal operational amplifier, a resistor and a condenser.

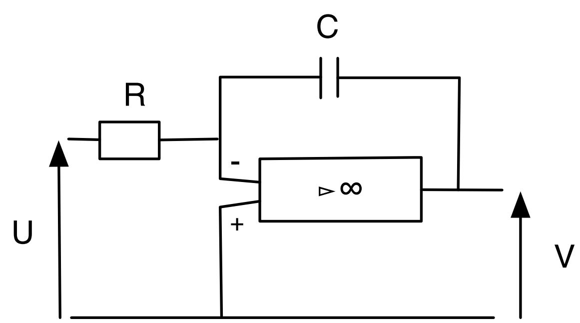


**Figure 3.** Realizing an integration with an ideal operational amplifier: one has $V(t) = -1/RC \int_0^t U(t)dt$

## 3 Planar mechanisms

The power of planar mechanisms made of rigid bars linked by their end by rivets attracted much attention in England and in France in the late 19th century, with a new birth of interest in Russia at the end of the forties: see for example [6], [26]. The pantograph, which allows to realize dilatations is well-known. The Peaucellier's mechanism allows transforming a linear motion into a circular motion.

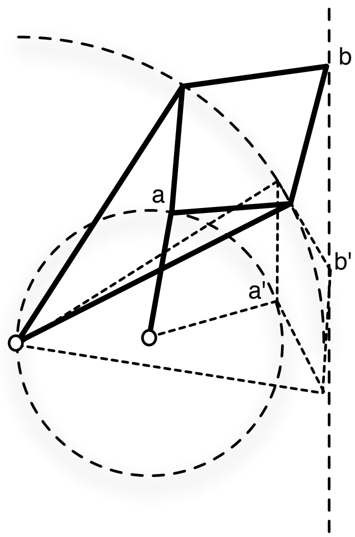


**Figure 4.** Peaucellier's mechanism. The circular motion of $a$ is transformed into a linear motion of $b$.

More generally, this is natural to ask what is the power of such devices. This is given by the following very nice result (see for e.g. [6], [26]) attributed to Kempe [18]: this corresponds to semi-algebraic sets.

**Theorem 2 (Completeness of planar mechanism).**

- *For any non-empty semi-algebraic set $S$, there exists a mechanism with $n$ points that move on linear segments, but that are free to move on these segments, and that forces the relation $(x_1, \ldots, x_n) \in S$, where $x_i$ are the distances on the linear segments.*
- *Conversely, the domain of evolution of any finite planar mechanism is semi-algebraic.*

# 4 Distributed Computations

## 4.1 Populations Protocols

We present the recent *population protocol model* of [2], proposed as a model for passively mobile sensor networks.

In this model, a protocol consists in giving a finite set of internal states $Q = \{1, 2, \ldots, k\}$, and transition rules given by $\delta : Q \times Q \to Q \times Q$. For $\delta(p, q) = (p', q')$, write $\delta_1(p, q) = p'$, $\delta_2(p, q) = q'$.

A configuration of a system at a given time is given by the internal states of each of the $n$ individuals.

We suppose that the individuals are completely indiscernible. It follows that the state of a system can be described by the number $n_i$ of individuals in state $i$, for $1 \leq i \leq k$, better than by the state of each individual.

At each discrete round, a unique individual $i$ is put in relation with some other individual $j$: at the end of this meeting, the individual $i$ is in state $\delta_1(q_i, q_j)$, and individual $j$ is in state $\delta_2(q_i, q_j)$.

We suppose that we cannot control the interactions, and that there is a notion of fairness: if in a configuration $C$ one can go to configuration $C'$ in one step (denoted by $C \to C'$) then in any derivation $C_0 C_1 \cdots$, with $C_i \to C_{i+1}$ for all $i$, if $C$ appears infinitely often, then $C'$ also.

One wants to consider population protocols as predicate recognizers $\psi : \mathbb{N}^m \to \{0, 1\}$.

To do so, fix a subset $Q^+ \subset Q$, and say that an tuple $(n_1, \ldots, n_m) \in \mathbb{N}^m$, for $m \leq k$, is accepted (respectively rejected) by the protocol, if starting from any configuration with $n_i$ individuals in state $i$, eventually all the individuals will be in some internal state that belongs to $Q^+$ (resp. its complement), and this stays true at any time afterward.

One says that the protocol recognizes $\psi : \mathbb{N}^m \to \{0, 1\}$ if for all tuple $(n_1, \ldots, n_m)$, it is accepted when $\psi(n_1, \ldots, n_m) = 1$ and it is rejected when $\psi(n_1, \ldots, n_m) = 0$.

We have the following very nice result (recall that the sets that are definable in Presburger arithmetic coincide with the semi-linear sets over the integers).

**Theorem 3 (Power of Population Protocols [4]).**

- *Any predicate* $\psi : \mathbb{N}^m \to \{0, 1\}$ *that can be defined in Presburger arithmetic can be computed by a population protocol.*
- *Conversely, any predicate* $\psi : \mathbb{N}^m \to \{0, 1\}$ *that is computed by a population protocol can be defined in Presburger arithmetic.*

For example, since this is definable in Presburger arithmetic, there is a protocol to decide if more than 5% of agents are in internal state 1.

This theorems shows, if needed, that these models are really different from classical models, such as cellular automata, or Turing machines.

Refer to [1], [5], [3] for more results about this model, and some variants.

### 4.2 Another Model

If the number of individuals is high, this is natural not to talk about numbers, but about proportions or statistics.

For example, consider the following protocol: we have a population of $n$ agents. Each agent is either in the state $+$, or in state $-$. Hence, a configuration corresponds to a point of $S = \{+, -\}^n$.

We suppose that time is discrete. At each discrete time (round), all (or a fixed fraction of) the agents interact in pairs, according to the following rules:

$$
\begin{aligned}
&++ \to 1/2+, 1/2- \\
&+- \to + \\
&-+ \to + \\
&-- \to 1/2+, 1/2-
\end{aligned}
$$

One must interpret the second rule in the following way: if an individual of type + interacts with an individual of type −, then it becomes of type +. One must interpret the first rule in the following way: if an individual is of type + interacts with an individual of type +, he becomes of type + with probability 1/2, and of type − with probability 1/2.

We suppose that the pairings are chosen at random uniformly.

Experimentally, the proportion of + in the population converges towards $\sqrt{2}/2$, when the number of individuals increases. This could be expected, since, if $p$ denotes the proportion of +, with probability $p$ an individual meets a +, and $1-p$ a −. Now, the first and fourth rule destroy in mean 1/2+ each, whereas the second and third rules create one + each. By doing the sum, one can write that in expectation, the number of + that are created at each round is

$$1/2p^2 + 2p(1-p) + 1/2(1-p)^2 = 1/2 + p - p^2.$$

Now, at equilibrium, there must be conservation, and so it must be equal to $p$. Hence $p^2 = 1/2$, i.e. $p = \sqrt{2}/2$.

The previous system converges towards $\sqrt{2}/2$, and hence can be considered as computing this value. Which numbers are computable by such protocols? Of course, by assuming using pairwise pairing, rational probabilities, and a finite number of internal states for each agent.

## 5 Computing with distributed games

More generally, in previous two models, rules of interactions can be considered as games between participants. When the number of individuals $n$ becomes high, models of dynamics of population, and of dynamics in game theory become natural and relevant. We review some of them.

### 5.1 Game Theory Models

To introduce some dynamism in game theory, there are two main approaches. The first consists in repeating games. The second in using models from evolutionary game theory.

Let's first present the simplest concepts from Game Theory [23]. We focus on non-cooperative games, with complete information, in extensive form.

The simplest game is a two player games, called $I$ and $II$, with a finite set of options, called *pure strategies*, $Strat(I)$ and $Strat(II)$. Denote by $a_{i,j}$ (respectively: $b_{i,j}$) the score (or if it is a random variable its expected value) for player $I$ (resp. $II$) when $I$ uses strategy $i \in Strat(I)$ and $II$ strategy $j \in Strat(II)$. The scores are given by $n \times m$ matrices $A$ and $B$, where $n$ and $m$ are the cardinality of $Strat(I)$ and $Strat(II)$.

*Example 1 (Prisonner dilemma).* The case where $A$ and $B$ are the following matrices

$$A = \begin{pmatrix} 3 & 0 \\ 5 & 1 \end{pmatrix}, B = \begin{pmatrix} 3 & 5 \\ 0 & 1 \end{pmatrix}$$

is called the *game of prisoners*, or *prisoner dilemma*. We denote by $C$ (for cooperation) the first pure strategy, and by $D$ (for defection) the second pure strategy of each player.

A *mixed strategy* of player $I$, which consists in using $i \in Strat(I)$ with probability $x_i$, will be denoted by vector $\mathbf{x} = (x_1, \ldots, x_n)^T$. One must have $\sum_{i=1}^{n} x_i = 1$, i.e. $\mathbf{x} \in S_n$, where $S_n$ is the unit simplex of $\mathbb{R}^n$, generated by the vectors $\mathbf{e}_i$ of the unit standard basis of $\mathbb{R}^n$. In a similar way, a mixed strategy for $II$ corresponds to $\mathbf{y} = (y_1, \ldots, y_m)^T$ with $\mathbf{y} \in S_m$. If player $II$ uses mixed strategy $\mathbf{x}$, and player $II$ mixed strategy $\mathbf{y}$, then the first has mean score $\mathbf{x}^T A\mathbf{y}$ and the second $\mathbf{x}^T B\mathbf{y}$. A strategy $\mathbf{x} \in S_n$ is said to be a best response to strategy $\mathbf{y} \in S_m$, denoted by $\mathbf{x} \in BR(\mathbf{y})$ if

$$\mathbf{z}^T A\mathbf{y} \leq \mathbf{x}^T Ay \tag{1}$$

for all strategy $\mathbf{z} \in S_n$. A pair $(\mathbf{x}, \mathbf{y})$ is a *mixed Nash equilibrium* if $\mathbf{x} \in BR(\mathbf{y})$ and $\mathbf{y} \in BR(\mathbf{x})$. Nash theorem [22] claims, by a fixed point argument, that such an equilibrium always exists. However, it is not necessarily unique.

### 5.2 Repeated Games

Repeating $k$ times a game is equivalent to extend the space of choices into $Strat(I)^k$ and $Strat(II)^k$: player $I$ (respectively $II$) chooses its action $\mathbf{x}(t) \in Strat(I)$, (resp. $\mathbf{y}(t) \in Strat(II)$) at time $t$ for $t = 1, 2, \cdots, k$. Hence, this is equivalent to a two-players game with respectively $n^k$ and $m^k$ choices for players.

In practice, player $I$ (respectively $II$) has to solve the following problem at each time (round) $t$: given the history of the game up to now, that is to say $X_{t-1} = \mathbf{x}(1), \cdots, \mathbf{x}(t-1)$ and $Y_{t-1} = \mathbf{y}(1), \cdots, \mathbf{y}(t-1)$ what should I play at time $t$? That is to say how to choose $\mathbf{x}(t) \in Strat(I)$? (resp. $\mathbf{y}(t) \in Strat(II)$?)

This is natural to suppose that the answer of each of the players is given by some behavior rules: $\mathbf{x}(t) = f(X_{t-1}, Y_{t-1})$, $\mathbf{y}(t) = g(X_{t-1}, Y_{t-1})$ for some functions $f$ and $g$. For example, the question of the best behavior rule to use for the prisoner lemma gave birth to an important literature, in particular, after the book [7].

### 5.3 Games on a Graph

An example of behavior for the prisoner lemma is $PAVLOV$.

*Example 2 ( $PAVLOV$ ).* The $PAVLOV$ behavior consists, in the iterated prisoner lemma, in fixing a threshold, say 3, and at time $t$, replaying the previous pure action if the last score is above this threshold, and changing the action otherwise.

Concretely, if we denote $+$ for $C$, and $-$ for $D$, one checks easily that this corresponds to rules

$$\begin{cases} ++ \rightarrow ++ \\ +- \rightarrow -- \\ -+ \rightarrow -- \\ -- \rightarrow ++, \end{cases} \tag{2}$$

where the left hand side of each rule denotes $\mathbf{x}(t-1)\mathbf{y}(t-1)$, and the right hand side the corresponding result for $\mathbf{x}(t)\mathbf{y}(t)$.

From a set of such rules, this is easy to obtain a distributed dynamic. For example, let's follow [11]: Suppose that we have a connected graph $G = (V, E)$, with $N$ vertices. The vertices correspond to players. An instantaneous configuration of the system is given by an element of $\{+, -\}^N$, that is to say by the state $+$ of $-$ of each vertex. Hence, there are $2^N$ configurations.

At each round $t$, one chooses randomly and uniformly one edge $(i, j)$ of the graph. At this moment, players $i$ and $j$ play the prisoner dilemma with the $PAVLOV$ behavior, that is to say the rules of the equation 2 are applied.

What is the final state reached by the system?

The underlying model is a huge Markov chain with $2^N$ states. The state $E^* = \{+\}^N$ is absorbing. If the graph $G$ does not have any isolated vertex, this is the unique absorbing state, and there exists a sequence of transformations that transforms any state $E$ into this state $E^*$. As a consequence, from well-known classical results for Markov chains, whatever the initial configuration is, with probability 1, the system will be in state $E^*$ [9]. The system is *self-stabilizing*. Several results about the convergence time towards this stable state can be found in [11], and [12], for rings, and complete graphs.

What is interesting in this example is that it shows how to go from a game, and behaviors to a distributed dynamic on a graph. Clearly this is easy to associate a similar dynamic to any[1] Markovian behavior on a symmetric game.

### 5.4 Myopic Dynamic

In the general case, to every 2-players repeated game, one can associate the *myopic* behavior. It consists in the fact that each player makes systematically the hypothesis that the opposite player will replay at time $t$ the same thing as he played at time $t-1$. As a consequence, this behavior consists in choosing systematically at time $t$ the (or a) best response to the action of the opposite player at time $t-1$:

$$f(X_{t-1}, Y_{t-1}) \in BR(\mathbf{y}(t-1)).$$

[1] But not necessarily Pavlovian. Actually, the behavior $PAVLOV$, as described here, is not ambiguous only on 2 by 2 matrices.

Take, like [8], the example of the Cournot duopoly game. The Cournot duopoly game is a well-known economical model of the competition of two producers of a same good. In this model, the production of a unit article of this good costs $c$. One makes the hypothesis that the total demand is of the form $q = q_1 + q_2 = M - p$, where $p$ is the sold price, and $q_1$ and $q_2$ the number of produced articles by each of the firms.

The problem of firm $I$ (respectively $II$) is to fix $q_1$ (resp. $q_2$) in order to maximize its profit $(p-c)q_1$ (resp. $(p-c)q_2$). One shows easily (see [8]), that the best response to $q_2$ is to choose $q_1 = 1/2(M-c-q_2)$, and that the best response to $q_1$ is to choose $q_2 = 1/2(M-c-q_1)$, so that the unique Nash equilibrium corresponds to the intersection of the two lines defined by these equations.

The myopic dynamic for the two players then gives on this game

$$\begin{cases} q_1(t) = 1/2(M - c - q_2(t-1)) \\ q_2(t) = 1/2(M - c - q_1(t-1)). \end{cases}$$

This is easy to show that whatever the initial point is, such a dynamic converges towards the Nash equilibrium. The collective dynamic converges towards the rational equilibrium. Unfortunately, as shown in [8], this is not always the case.

### 5.5 Fictious Player Dynamic

The myopic behavior can be considered as very too basic. A more reasonable behavior seems to be the following: to predict what will play the opposite player at time $t$, let's use the statistic of what he did at time $1, 2, \cdots, t-1$: if he played action $i$ $n_i$ times, let's estimate that he will play action $i$ with probability $x_i = n_i/(t-1)$ at time $t$. This is what is called the *fictious player dynamic*.

To simplify things, let's follow [8], and suppose that $n = m = 2$, and that the matrices are given by

$$A = \begin{pmatrix} 0 & 3 \\ 2 & 1 \end{pmatrix}, \; B = \begin{pmatrix} 2 & 0 \\ 1 & 3 \end{pmatrix}.$$

If at time $1, 2, \cdots, t-1$, player 2 used $n_i$ times action number $i$, player $I$ will estimate that player $II$ will play at time $t$ action $i$ with probability $y_i(t) = n_i/(t-1)$. Player $II$ will evaluate probability $x_i(t)$ that player $I$ play action $i$ in a symmetric way.

To study the dynamic, as this is shown in [8], one just needs to go from discrete time to continuous time: a simple analysis (see [8]) shows that as long as $(x_2(t), y_2(t))$ stays in zone $A$ of the left part of Figure 5, player $I$ will use its second pure strategy, and player $II$ its first pure strategy as a best response to what he or she expects from the opposite player.

The dynamic $(x_2(t), y_2(t))$ will stay in this zone up to time $t + \tau$ for $\tau > 0$ sufficiently small. Since we know the choice of player $II$ between time $t$ and time $t + \tau$, one can hence evaluate $y_2(t + \tau)$ as

$$y_2(t+\tau) = \frac{t y_2(t)}{t+\tau}. \tag{3}$$

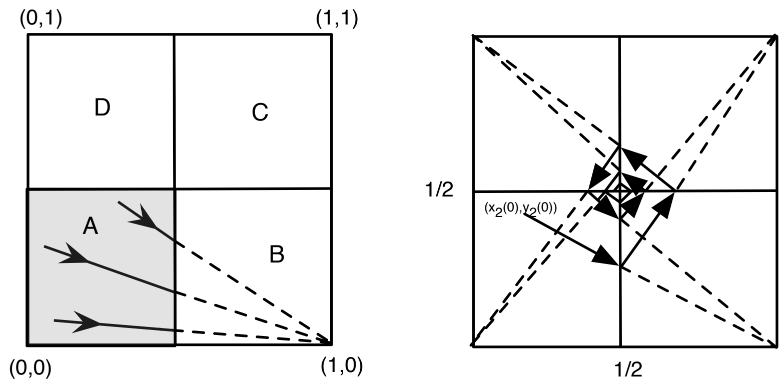


**Figure 5.** Convergence towards a mixed equilibrium.

This can be written as $\frac{y_2(t+\tau)-y_2(t)}{\tau} = -y_2(t)$.

Bet letting $\tau$ converge to 0, we obtain $y_2'(t) = \frac{y_2(t)}{t}$.

In a similar way, we obtain $x_2'(t) = \frac{1-x_2(t)}{t}$.

The points that satisfy these two equations are on a straight line that starts from $(x_2(t), y_2(t))$ and that joins point $(1,0)$. A similar study on zones $B$,$C$, and $D$ of the left part of Figure 5 shows that the dynamic must be the one depicted on the right part of Figure 5. It converges towards the mixed Nash equilibrium of the game. Once again, the collective dynamic converges towards the rational equilibrium. Unfortunately, once again, this is not the case for all the games: one can easily consider games where trajectories do not converge, or with limit cycles [8].

### 5.6 Evolutionary Game Theory Models

Evolutionary game theory is another way to associate dynamics to games.

Evolutionary game theory is born from the book from Maynard Smith [20].

To illustrate how, to a game, can be associated a biological dynamic, let's take the fictive example of a population of individuals from [8]. Binmore chooses to call these individuals *dodos*.

The day of a dodo lasts a fraction $\tau$ of a year. There are $n$ types of dodos: the dodos that play action 1, the dodos that play action 2, ..., and the dodos that play action $n$. Babies of a dodo of type $i$ are always of type $i$.

We are interested in the proportion $x_i(t)$ of dodos that play action $i$. We have of course $\sum_{i=1}^{n} x_i(t) = 1$.

At the end of each day, the dodos fight pairwise. The outcome of a fight has some influence on the fecundity of involved participants. One reads on a matrix $A$ $n \times n$ at entry $a_{i,j}$ the birth rate of a given dodo, if it is of type $i$ and if he fights again an individual of type $j$: his expected number of babies at next day is given by $\tau a_{i,j}$.

How many babies a dodo of type $i$ can expect to have at next day? The answer is

$$\sum_{j=1}^{n} x_j(t)\tau a_{i,j} = (A\mathbf{x})_i\tau.$$

Indeed, since the pairing for the fights on the evening between dodos are chosen at random and uniformly among the population, in expectation its birth rate is given by previous expression.

The number of dodos of type $i$ in the next morning is hence

$$Nx_i(t)(1 + (A\mathbf{x})_i\tau).$$

Mortality being not a function of the type of dodos, the next day, the fraction of dodos of type $i$ will be given by

$$x_i(t+\tau) = \frac{Nx_i(t)(1 + (A\mathbf{x})_i\tau)}{Nx_1(t)(1 + (A\mathbf{x})_1\tau) + \cdots + Nx_n(t)(1 + (A\mathbf{x})_n\tau)}.$$

Hence

$$x_i(t+\tau) = \frac{x_i(t)(1 + (A\mathbf{x})_i\tau)}{1 + \mathbf{x}^T A\mathbf{x}\tau}.$$

where $\mathbf{x}^T A\mathbf{x}\tau$ can be interpreted as the expected number of birth for a dodo in a day.

This can be rewritten as

$$\frac{x_i(t+\tau) - x_i(t)}{\tau} = x_i(t)\frac{(A\mathbf{x})_i - \mathbf{x}^T A\mathbf{x}}{1 + \mathbf{x}^T A\mathbf{x}\tau}.$$

By taking limit when $\tau$ goes to 0, we obtain

$$x_i' = x_i((A\mathbf{x})_i - \mathbf{x}^t A\mathbf{x}).$$

This is what is called *replicator dynamic*. Such an equation models the fact that individuals whose score (fitness) given by matrix $A$ is above mean score (fitness) have tendency to reproduce, whereas those that have a score under the mean score have tendency to disappear.

Of course the model about dodos is relatively ad hoc, but many situations and models give rise to same dynamics: see e.g. [20].

Evolutionary game theory aims at studying the behaviors of such dynamics in function of matrix $A$.

It has its own notions of equilibria, motivated by the stability of underlying dynamical systems, such as the notion of evolutionary stable equilibrium. An evolutionary stable equilibrium is a particular Nash equilibrium. It makes it possible to link the notions of equilibria for the game given by $A$ to the notions of stability for the corresponding dynamical system.

Actually, it does not only consider replicator dynamics, but other dynamics such as imitation dynamics, best response dynamics, and so on. . . , with in all dynamics the idea that individuals with highest score reproduce faster than others. Refer to [20], [27], [17] for presentations.

## References


1. Dana Angluin, James Aspnes, Melody Chan, Michael J. Fischer, Hong Jiang, and René Peralta. Stably computable properties of network graphs. In Viktor K. Prasanna, Sitharama Iyengar, Paul Spirakis, and Matt Welsh, editors, *Distributed Computing in Sensor Systems: First IEEE International Conference, DCOSS 2005, Marina del Rey, CA, USE, June/July, 2005, Proceedings*, volume 3560 of *Lecture Notes in Computer Science*, pages 63–74. Springer-Verlag, June 2005.
2. Dana Angluin, James Aspnes, Zoë Diamadi, Michael J. Fischer, and René Peralta. Computation in networks of passively mobile finite-state sensors. In *Twenty-Third ACM Symposium on Principles of Distributed Computing*, pages 290–299. ACM Press, July 2004.
3. Dana Angluin, James Aspnes, and David Eisenstat. Fast computation by population protocols with a leader. In *20th International Symposium on Distributed Computing (DISC'2006)*, Lecture Notes in Computer Science. Springer, 2006. To appear.
4. Dana Angluin, James Aspnes, and David Eisenstat. Stably computable predicates are semilinear. In *PODC '06: Proceedings of the twenty-fifth annual ACM symposium on Principles of distributed computing*, pages 292–299, New York, NY, USA, 2006. ACM Press.
5. Dana Angluin, James Aspnes, Michael J. Fischer, and Hong Jiang. Self-stabilizing population protocols. In *Ninth International Conference on Principles of Distributed Systems (OPODIS'2005)*, Lecture Notes in Computer Science, pages 79–90. Springer, December 2005. To appear.
6. I.I. Artobolevskii. *Mechanisms for the generation of plane curves*. Macmillan, New York, 1964. Translated by R.D. Wills and W. Johnson.
7. Robert M. Axelrod. *The Evolution of Cooperation*. Basic Books, 1984.
8. Ken Binmore. *Jeux et Théorie des jeux*. DeBoeck Université, Paris-Bruxelles, 1999. Translated from 'Fun and Games: a text on game theory" by Francis Bismans and Eulalia Damaso.
9. Pierre Brémaud. *Markov Chains, Gibbs Fields, Monte Carlo Simulation, and Queues*. Springer-Verlag, New York, 2001.
10. V. Bush. The differential analyser. *Journal of the Franklin Institute*, 212(4):447–488, 1931.
11. Martin E. Dyer, Leslie Ann Goldberg, Catherine S. Greenhill, Gabriel Istrate, and Mark Jerrum. Convergence of the iterated prisoner's dilemma game. *Combinatorics, Probability & Computing*, 11(2), 2002.
12. Laurent Fribourg, Stéphane Messika, and Claudine Picaronny. Coupling and self-stabilization. In Rachid Guerraoui, editor, *Distributed Computing, 18th International Conference, DISC 2004, Amsterdam, The Netherlands, October 4-7, 2004, Proceedings*, volume 3274 of *Lecture Notes in Computer Science*, pages 201–215. Springer, 2004.
13. Daniel S. Graça. Some recent developments on Shannon's General Purpose Analog Computer. 50(4-5):473–485, 2004.

14. Daniel S. Graça. *Computability with Polynomial Differential Equations*. PhD thesis, Instituto Superior Técnico, 2007.
15. Daniel S. Graça and José Félix Costa. Analog computers and recursive functions over the reals. *Journal of Complexity*, 19(5):644–664, 2003.
16. Morris W. Hirsch, Stephen Smale, and Robert Devaney. *Differential Equations, Dynamical Systems, and an Introduction to Chaos*. Elsevier Academic Press, 2003.
17. J. Hofbauer and K. Sigmund. Evolutionary game dynamics. *Bulletin of the American Mathematical Society*, 4:479–519, 2003.
18. A.B. Kempe. On a general method of describing plane curves of the n–th degree by linkwork. *Proceedings of the London Mathematical Society*, 7:213–216, 1876.
19. L. Lipshitz and L. A. Rubel. A differentially algebraic replacement theorem, and analog computability. *Proceedings of the American Mathematical Society*, 99(2):367–372, February 1987.
20. J. Maynard-Smith. *Evolution and the Theory of Games*. Cambridge University Press, Cambridge, 1981.
21. James Dickson Murray. *Mathematical Biology. I: An Introduction*. Springer, third edition, 2002.
22. John F. Nash. Equilibrium points in $n$-person games. *Proc. of the National Academy of Sciences*, 36:48–49, 1950.
23. Martin J. Osbourne and Ariel Rubinstein. *A Course in Game Theory*. MIT Press, 1994.
24. M. B. Pour-El. Abstract computability and its relation to the general purpose analog computer (some connections between logic, differential equations and analog computers). *Transactions of the American Mathematical Society*, 199:1–28, 1974.
25. C. E. Shannon. Mathematical theory of the differential analyser. *Journal of Mathematics and Physics MIT*, 20:337–354, 1941.
26. Antonin Svoboda. *Computing Mechanisms and Linkages*. McGraw Hill, 1948. Reprinted by Dover Publications in 1965.
27. Jörgen W. Weibull. *Evolutionary Game Theory*. The MIT Press, 1995.