A Multiagent System Generating Complex Behaviours

Florin Leon

Department of Computer Science and Engineering, Technical University of Iaşi,
Bd. Mangeron 27, 700050 Iaşi, Romania
fleon@cs.tuiasi.ro

Abstract. In this paper we describe the design of a multiagent system based on simple interaction rules that can generate different overall behaviours, from asymptotically stable to chaotic, verified by the corresponding largest Lyapunov exponent. We show that very small perturbations can have a great impact on the evolution of the system, and we investigate some methods of controlling such perturbations in order to have a desirable final state.

Keywords: multiagent system, chaotic behaviour, perturbations, chaos control.

1 Introduction

Chaos has been extensively studied in physical systems, including methods to control it for uni-, bi- and multi-dimensional systems [1]. Also, concepts such as causality and the principle of minimal change in dynamic systems have been formalized [11].

Many human-related e.g. social or economic systems are nonlinear, even when the underlying rules of individual interactions are known to be rational and deterministic. Prediction is very difficult or impossible in these situations. However, by trying to model such phenomena, we can gain some insights regarding the fundamental nature of the system. Surprising or counterintuitive behaviours observed in reality can be sometimes explained by the results of simulations.

Therefore, the emergence of chaos out of social interactions is very important for descriptive attempts in psychology and sociology [7], and multiagent systems are a natural way of modelling such social interactions. Chaotic behaviour in multiagent systems has been investigated from many perspectives: the control of chaos in biological systems with a map depending on growth rate [13], the use of a chaotic map by the agents for optimization [2] and image segmentation [10], or the study of multiagent systems stability for economic applications [3]. However, in most of these approaches, chaos is explicitly injected into the system, by using a chaotic map, e.g. the well-known logistic map, in the decision function of the agents.

The main goal of this work is the design of simple interaction rules which in turn can generate, through a cascade effect, different types of overall behaviours, from stable to chaotic. We believe that these can be considered metaphors for the different kinds of everyday social or economic interactions, whose effects are sometimes entirely predictable and can lead to an equilibrium while some other times fluctuations can widely affect the system state, and even if the system appears to be

C. Bădică, N.T. Nguyen, and M. Brezovan (eds.): ICCCI 2013, LNAI 8083, pp. 154–164, 2013.
© Springer-Verlag Berlin Heidelberg 2013

stable for long periods of time, sudden changes can occur unpredictably because of subtle changes in the internal state of the system. We also aim at investigating how very small changes can non-locally ripple throughout the system with great consequences and if it is possible to reverse these changes in a non-trivial way, i.e. by slightly adjusting the system after the initial perturbation has occurred.

The paper is organized as follows. Section 2 presents the interaction protocol of the multiagent system and its mathematical formalization. Section 3 discusses the deterministic and chaotic behaviours that emerge from the system execution. Section 4 presents an experimental study regarding the effects of small perturbations in the initial state of the system and the possibility of cancelling them through minimal external interventions. The final section contains the conclusions of this work.

2 The Design of the Multiagent System

The main goal in designing the structure and the interactions of the multiagent system was to find a simple setting that can generate complex behaviours. A delicate balance is needed in this respect. On the one hand, if the system is too simple, its behaviour will be completely deterministic and predictable. On the other hand, if the system is overly complex, it would be very difficult to assess the contribution of the individual internal elements to its observed evolution. The multiagent system presented as follows is the result of many attempts of finding this balance.

The proposed system is comprised of n agents; let A be the set of agents. Each agent has m needs and m resources, whose values lie in their predefined domains $D_n, D_r \subset \mathbb{R}^+$. This is a simplified conceptualization of any social or economic model, where the interactions of the individuals are based on some resource exchanges, of any nature, and where individuals have different valuations of the types of resources involved.

In the present model, it is assumed that the needs of an agent are fixed (although an adaptive mechanism could be easily implemented, taking into account, for example, previous results [8,9]), that its resources are variable and they change following the continuous interactions with other agents.

Also, the agents are situated in their execution environment: each agent a has a position π_a and can interact only with the other agents in its neighbourhood Λ_a. For simplicity, the environment is considered to be a bi-dimensional square lattice, but this imposes no limitation on the general interaction model – it can be applied without changes to any environment topology.

2.1 Social Model

Throughout the execution of the system, each agent, in turn, chooses another agent in its local neighbourhood to interact with. Each agent a stores the number of previous interactions with any other agent b, $i_a(b)$, and the cumulative outcome of these interactions, $o_a(b)$, which is based on the profits resulted from resource exchanges, as described in the following section.

When an agent a must choose another agent to interact with, it chooses the agent in its neighbourhood with the highest estimated outcome: $b^* = \arg\max_{b \in \Lambda_a} o_a(b)$.

The parallelism of agent execution is simulated by running them sequentially and in random order. Since one of the goals of the system is to be deterministic, we define the execution order from the start. Thus, at any time, it can be known which agent will execute and which other agent it will interact with. When perturbations are introduced into the system, the same execution order is preserved. It has been shown that the order of asynchronous processes plays a role in self-organisation within many multi-agent systems [4]. However, in our case this random order is not necessary to generate complex behaviours. Even if the agents are always executed in lexicographic order (first A1, then A2, then A3 etc.), sudden changes in utilities still occur, although the overall aspect of the system evolution is much smoother.

2.2 Bilateral Interaction Protocol

In any interaction, each agent tries to satisfy the needs of the other agent as well as possible, i.e. in decreasing order of its needs. The interaction actually represents the transfer of a resource quantum γ from an agent to the other. Ideally, each agent would satisfy the greatest need of the other.

For example, let us consider 3 needs (N) and 3 resources (R) for 2 agents a and b: $N_a = \{1, 2, 3\}$, $N_b = \{2, 3, 1\}$, $R_a = \{5, 7, 4\}$, $R_b = \{6, 6, 5\}$, and $\gamma = 1$. Since need 2 is the maximum of agent b, agent a will give b 1 unit of resource 2. Conversely, b will give a 1 unit of resource 3.

In order to add a layer of nonlinearity, we consider that an exchange is possible only if the amount of a resource exceeds a threshold level θ and if the giving agent a has a greater amount of the corresponding selected resource r_{sel} than the receiving agent b: $R_a(r_{sel}) > R_b(r_{sel})$ and $R_a(r_{sel}) > \theta$.

In the previous situation, if we impose a threshold level $\theta = 5$, agent a will still give b 1 unit of resource 2, but b will only satisfy need 1 for agent a.

Based on these exchanges, the resources are updated and the profit p_a is computed for an agent a as follows:

$$p_a = \gamma \cdot N_a(r_{sel}) \cdot \frac{R_b(r_{sel})}{R_a(r_{sel})}. \quad (1)$$

A bilateral interaction can bring an agent a profit greater or equal to 0. However, its utility should be able to both increase and decrease. For this purpose, we can compute a statistical average of the profit, p_{avg}, and increase the utility of an agent if the actual profit is above p_{avg}, and decrease the utility if the profit is below p_{avg}.

Thus, the equation for updating the utility level of an agent a is:

$$u_a \leftarrow \frac{u_a \cdot i_a^{adj} + \eta \cdot \left(p_a - p_{avg}\right)}{i_a^{adj} + 1}, \quad (2)$$

where the adjusted number of interactions is: $i_a^{adj} = \min\left(\sum_{b\in A} i_a(b), i_{mem}\right)$, i_{mem} is the maximum number of overall interactions that the agent can "remember" (i.e. take into account) and η is the rate of utility change. At the beginning, the utility of the agent can fluctuate more, as the agent explores the interactions with its neighbours. Afterwards, the change in utility decreases, but never becomes too small.

For example, if i_{mem} = 20, u_a = 0.1, p_a = 8.5, η = 1, p_{avg} = 7.5 and the sum of all previous interactions is 2, the utility will change to: u_a' = (0.1 · 2 + (8.5 – 7.5) · 1) / 3 = 0.4. If the sum of all previous interactions is 100, the same utility will change only to: u_a' = (0.1 · 20 + (8.5 – 7.5) · 1) / 21 = 0.14.

Similarly, the social outcome of an agent *a* concerning agent *b* is updated as follows:

$$o_a(b) \leftarrow \frac{o_a(b) \cdot i_a(b) + \eta \cdot \left(p_a - p_{avg}\right)}{i_a(b) + 1}. \quad (3)$$

In this case, the social model concerns only 1 agent and thus the use of the actual number of interactions can help the convergence of the estimation an agent has about another.

Regarding the computation of the average profit, we used a statistical approach where we took into account 100 continuous interactions between two randomly initialized agents, which exchange resources for 100000 time steps. The average profit depends on the number of resources, their domain and the interaction threshold.

3 Types of Behaviours

A key challenge in applied dynamical systems is the development of techniques to understand the internal dynamics of a nonlinear system, given only its observed outputs [5]. As the observed output of our multiagent system, we consider only the agent utilities. We can view this output as a discrete time series, one for each agent. In the following, we analyse the evolution of these time series over time. Since there is no stopping condition for the agent interactions, we restrict our study to a predefined, finite time horizon, e.g. 1000, 2000 or 10000 time steps.

Depending on the number of agents and the initial state of the system, several types of behaviours can be observed:

- *Asymptotically stable:* When only 2 agents exist in the system, we noticed that they can perform an indefinite number of interactions. They can stabilize to a continuous exchange of resources, possibly the same resource in both cases (γ units of the same resource are passed back and forth between the 2 agents). With 2 agents, the system quickly converges to a stable state (figure 1). Depending on the initial state, a stable state can also be reached by some agents in a system with multiple agents. The typical behaviour in the latter case is a high frequency vibration around the value of convergence. However, it is also possible that multiple agents all converge to stable states and the system remains in equilibrium afterwards;

- *Quasiperiodic:* With more interacting agents in the system, usually their utilities no longer converge to a stable value. Instead, the values belong to a certain range, with few, predictable oscillations around the mean value. Figure 2 shows the evolution of the utility of 4 agents over 10000 time steps. In order to smooth out short-term fluctuations and highlight longer-term trends, a simple moving average method is used, with a window size of 10 time steps;
- *Chaotic:* With a high number of agents (e.g. over 10), the complexity of their interactions usually exceeds the deterministically predictable level. The utilities of some agents widely fluctuate, even after the initial period where a part of the system approaches a stable zone. Figure 3 displays the behaviour of 100 agents over 10000 time steps. A simple moving average is applied here again, with a window size of 100 time steps. One agent (with a utility value around -3) has unpredictable great changes, although they appear to be governed by a higher-level order of some kind. Another agent has a sudden drop in utility around time step 9000, although it has been fairly stable before.

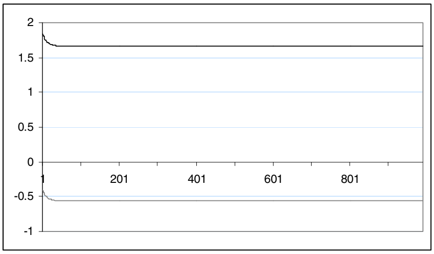

Fig. 1. Asymptotically stable behaviour - 2 agents, 1000 time steps

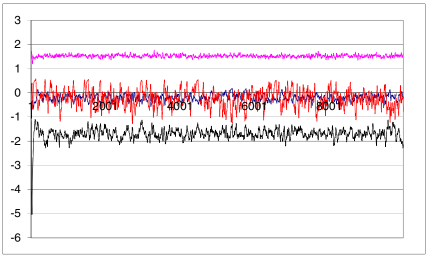

Fig. 2. Quasiperiodic behaviour - 4 agents, 10000 time steps

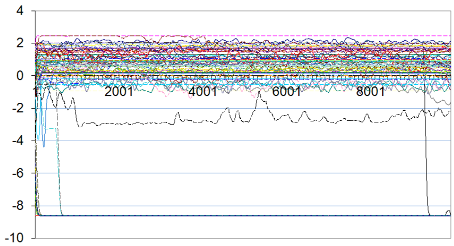

Fig. 3. Chaotic behaviour - 100 agents 10000 time steps

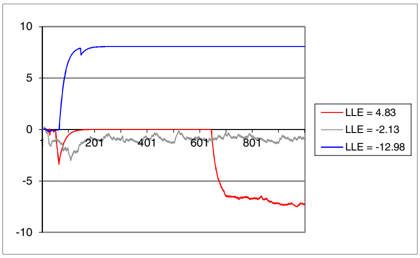

Fig. 4. Chaotic and non-chaotic variations

We consider that the third type of behaviour is chaotic, since it satisfies the typical features of chaos [6]:

- *Nonlinearity:* Given the nonlinearity caused by the minimum threshold for resource exchange, the system can be viewed as a hybrid one, with transitions between different ways of operation. Also, the maximum of the social outcome can change, thus an agent can interact with different neighbours, which results in different profits and further changes to the social outcomes;
- *Determinism:* Apart from the random initialization of the agent parameters (which nevertheless can be controlled by using the same seed for the random number generator), all the interaction rules are deterministic;
- *Sensitivity to initial conditions:* As we will show in the experimental study presented in section 4, very small changes in the initial state of the system can lead to a radically different final state;
- *Sustained irregularity and mostly impossible long-term predictions:* These are also characteristic of observed behaviours.

Regarding the effect of small perturbations, which in general can be used to control a chaotic system, out of many runs under different configurations, we noticed that a perturbation can affect the overall system behaviour in more ways:

- *No effect* within a predefined time horizon: depending on the agent positions, the system state and the place where the perturbation occurs, some changes can have no effect at all;
- *A temporary effect* which is later cancelled out within the time horizon;
- *A permanent effect* which reflects in the final state of the system, within the predefined time horizon.

We can make a parallel between these kinds of effects and the choices we make in everyday life. Out of the many alternatives that we have, a different choice can have no effect or sometimes we may not know that something is different until a later time when the different choice becomes relevant. Other times, a different choice impacts our environment immediately. Even if something changes, the overall environment can eventually reduce the perturbation, or the system can toggle to a whole different state indefinitely. All these kinds of behaviours have been observed in the designed multiagent system.

We can measure the degree of chaos introduced by a perturbation by considering the difference between the changed system and the original system as a time series, and computing the largest Lyapunov exponent (LLE) of the variation in an agent utility. Basically, LLE describes the predictability of a dynamical system. A positive value usually indicates that the system is chaotic [12]. There are methods, e.g. [14], which compute the LLE from the output of the system regarded as a time series.

Figure 4 displays three situations. The variation with a positive LLE (4.83) can be considered to be chaotic. We can notice the sudden change in utility after the half of the simulation, although the perturbation has occurred in the first time step. A small negative LLE (-2.13) indicates an almost deterministic behaviour, which can correspond to a quasiperiodic variation. Finally, a high negative LLE (-12.98) indicates a deterministic behaviour, when the time series converges to a value and remains stable there. Positive LLEs are not only found in some utility variations, but also in some of the original utility evolutions, depending on the system initial state.

4 Experimental Studies

A mathematical analysis of a nonlinear hybrid system is usually very difficult. Therefore, in the following, we will present an empirical experimental study, where we will emphasise different cases or settings which reveal certain types of behaviour.

Since one of the characteristics of a chaotic system is that small changes in its initial state can greatly affect the final state through a cascade effect, we observe the influence of perturbations on the system behaviour. We also reflect on the question of when it is possible to correct some distortions with the smallest amount of external energy, such that, after a perturbation, the system should reach again a desired state within a corresponding time horizon, through small changes.

In all the case studies presented in this section, the following parameters were used: the number of agents n = 10, the number of needs and resources m = 10, their domains $D_n = D_r$ = [0, 10), the resource transfer quantum γ = 1, the resource exchange threshold θ = 5, the interaction memory i_{mem} = 20, the utility change rate η = 2, the side length of the agent square neighbourhood Λ is 4 and the computed average profit p_{avg} = 7.5.

4.1 Original Behaviour

The configuration under study is composed of 3 subgraphs (figure 5): one agent, A1, is isolated and cannot interact with any other agent. Two agents, A2 and A3, form their own bilateral subsystem and seven agents can interact with one another in their corresponding neighbourhoods. A change in any of those agents can affect any other one in this subgraph, because, for example, A4 can influence A7, A7 can influence A9 and A9 can influence A10. The evolution of the agent utilities for 2000 time steps is displayed in figure 6.

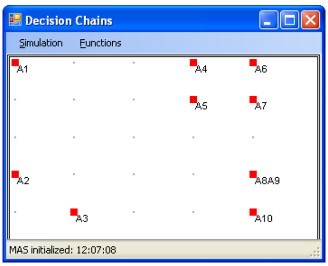

Fig. 5. The positions of the agents

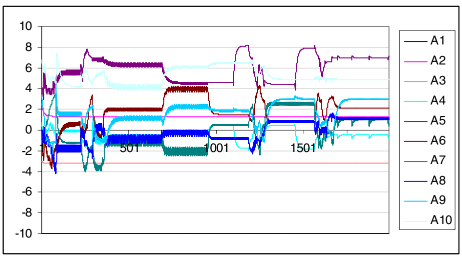

Fig. 6. The original evolution of agent utilities with no perturbation

4.2 The Effect of Small Perturbations

In this section, we observe the evolution of the utilities when a very small perturbation is added to or subtracted from a resource of an agent. Figure 7 shows the difference between the changed behaviour due to the presence of the perturbation and the original behaviour seen in figure 6, with a slightly larger perturbation of 0.1, and when the agents execute in lexicographic order. Figure 8 shows this difference for a perturbation of only 10^{-5} and when agents execute in a predefined random order. With 10 agents and 10 resources, this corresponds to a 10^{-7} change in the initial system state. We can see that, in general, the smaller a perturbation is, the longer it takes for its effect to accumulate and impact the observed behaviour of the system.

The actual number of an agent or resource is not very important, as we study the overall performance of the system. However, one can notice that the effects are non-local, and a change in one agent can affect other agents in its subgraph. Also, even if the perturbation has occurred in the first time step, big differences can appear later on, after 686 and 1873 time steps, respectively.

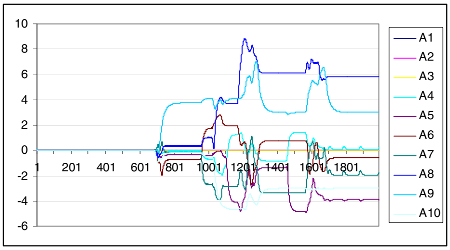

Fig. 7. The consequences of a perturbation of 0.1 in resource 5 of agent A3

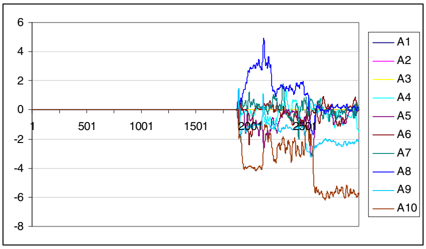

Fig. 8. The consequences of a perturbation of -10^{-5} in resource 6 of agent A6

4.3 Perturbation Correction

Given a perturbation in the initial time step with a non-null effect on the system, we are interested in finding a way to cancel or greatly reduce its impact, as observed on the time horizon and even beyond. Since this correction must be done from outside the system, and consists in changing the amount of a resource of an agent, it is also important that we find the minimum (or a small) amount of change needed to return the system to its final state as it would have been with no perturbation.

We would also like to find flexible solutions. A trivial solution would be to reverse the perturbation in the first time step. However, it is more interesting to see if there can be changes in later steps of the simulation which can tackle the effect of the initial perturbation.

Because the effects of change are non-local and can propagate throughout the subgraph of an agent's neighbours, we have applied, so far, the following search methods:

- *Exhaustive search with one correction point:* trying all the resources of all the agents in each step of the simulation, adding or subtracting a small amount (e.g. 0.1, 0.5), and observing the maximum utility variation in the final state of the system. If this maximum variation is below a desired threshold (e.g. 1), then a solution has been found;
- *Random corrections with one or multiple points:* considering 1 or more (e.g. 3) sets of quadruples (agent, resource, simulation step, correction amount) which inject changes into the simulation, and seeing if the final state of the system matches the final state in the original setting. The random search is by far faster than exhaustive search, but it cannot tell if any solution exists at all.

Besides considering only the state of the system at the time horizon (e.g. 2000 time steps), it is also important to verify if the system behaviour continues to be desirable. Figure 9 shows the effect of a 1 point correction for the situation presented in figure 7, which remains stable for a test period of 100 more time steps after the initial 2000 ones. However, if the system is chaotic, it is impossible to guarantee that this difference will remain small forever.

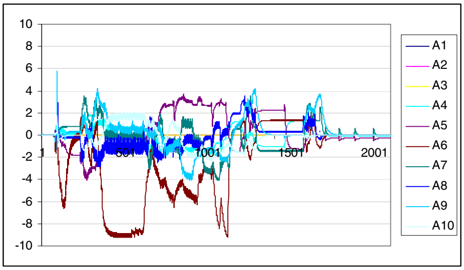

Fig. 9. A perturbation correction with an amount of -0.5 in resource 2 of agent A4 in step 70, leading to a maximum difference of 0.48 utility units from the original final state within the test period of 100 time steps

5 Conclusions

In this paper we presented the design of a multiagent system that can display different types of behaviours, from asymptotically stable to chaotic. In this case, chaos arises only from the agent interactions, and it is not artificially introduced through a chaotic map.

As future directions of research, we aim at further analysing the results of the interactions in order to see whether some probabilistic predictions can be made, taking into account the system state at a certain moment. It is important to determine when small perturbations have visible effects and when they can be controlled. Also, one must investigate whether classical chaos control techniques used for physical systems such as the OGY method, can be applied as well for this multiagent system.

Another fundamental question is whether the chaos in the system is only transient and eventually stabilises into a steady state or its behaviour remains chaotic forever.

Out of many experiments, it seems that sometimes the system converges to a stable state. In other cases, chaos doesn't seem to be only transient, e.g. with 50 agents executing in lexicographic order (which corresponds to fewer fluctuations), there are still sudden changes occurring in the utility variation even after 50000 time steps. One needs to distinguish between these cases as well.

So far, the proposed system is mainly of theoretical importance, but one can investigate if it can be used for social or economic simulations, for the modelling of biological processes or other typical applications.

Acknowledgment. This work was supported by the "Partnership in priority areas – PN-II" program, financed by ANCS, CNDI - UEFISCDI, project PN-II-PT-PCCA-2011-3.2-0732, No. 23/2012.

References

1. Boccaletti, S., Grebogi, C., Lai, Y.C., Mancini, H., Maza, D.: The Control of Chaos: Theory and Applications. Physics Reports 329, 103–197 (2000)
2. Charrier, R., Bourjot, C., Charpillet, F.: A Nonlinear Multi-agent System designed for Swarm Intelligence: The Logistic MAS. In: First IEEE International Conference on Self-Adaptive and Self-Organizing Systems, pp. 32–44 (2007)
3. Chli, M., De Wilde, P., Goossenaerts, J., Abramov, V., Szirbik, N., Correia, L., Mariano, P., Ribeiro, R.: Stability of Multi-Agent Systems. In: IEEE International Conference on Systems, Man and Cybernetics, vol. 1, pp. 551–556 (2003)
4. Cornforth, D., Green, D.G., Newth, D.: Ordered asynchronous processes in multi-agent systems. Physica D (Nonlinear Phenomena) 204, 70–82 (2005)
5. Dawes, J.H.P., Freeland, M.C.: The '0–1 test for chaos' and strange nonchaotic attractors, http://people.bath.ac.uk/jhpd20/publications/sna.pdf (preprint)
6. Ditto, W., Munakata, T.: Principles and Applications of Chaotic Systems. Communications of the ACM 38(11), 96–102 (1995)
7. Katerelos, I.D., Koulouris, A.G.: Is Prediction Possible? Chaotic Behavior of Multiple Equilibria Regulation Model in Cellular Automata Topology, Complexiy, Wiley Periodicals 10(1) (2004)
8. Leon, F.: Multiagent Role Allocation Based on Cognitive Dissonance Theory. International Review on Computers and Software 6(5), 715–724 (2011)
9. Leon, F.: Self-Organization of Roles Based on Multilateral Negotiation for Task Allocation. In: Klügl, F., Ossowski, S. (eds.) MATES 2011. LNCS, vol. 6973, pp. 173–180. Springer, Heidelberg (2011)
10. Melkemi, K.E., Batouche, M., Foufou, S.: Chaotic MultiAgent System Approach for MRF-based Image Segmentation. In: Proceedings of the 4th International Symposium on Image and Signal Processing and Anlysis, ISPA 2005, Zagreb, Croatia, pp. 268–273 (2005)
11. Pagnucco, M., Peppas, P.: Causality and Minimal Change Demystified. In: Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI 2001), Seattle, USA, pp. 125–130 (2001)

12. Rothman, D.: Nonlinear Dynamics I: Chaos (2006), http://ocw.mit.edu/courses/earth-atmospheric-and-planetary-sciences/12-006j-nonlinear-dynamics-i-chaos-fall-2006/lecture-notes/lecnotes15.pdf
13. Solé, R.V., Gamarra, J.G.P., Ginovart, M., López, D.: Controlling Chaos in Ecology: From Deterministic to Individual-based Models. Bulletin of Mathematical Biology 61, 1187–1207 (1999)
14. Wolf, A., Swift, J.B., Swinney, H.L., Vastano, J.A.: Determining Lyapunov exponents from a time series. Physica D (Nonlinear Phenomena) 16, 285–317 (1985)