
Article

Synchronization, control and data assimilation of the Lorenz system

Franco Bagnoli ^{1,2,†} * and Michele Baia ^{1,†}

¹ Department of Physics and Astronomy and CSDC, University of Florence, via G. Sansone 1, 50019 Sesto Fiorentino (Italy); {franco.bagnoli,michele.baia}@unifi.it

² INFN, Sect. Florence.

* Correspondence: franco.bagnoli@unifi.it (F.B.)

† These authors contributed equally to this work.

Abstract: We explore several aspects of replica synchronization with the goas of retrieving the value of parameters for the Lorenz system. The idea is that of having a computer replica (slave) of a natural system (master, simulated in this paper), and exploit the fact that slave synchronizes with the master only if they evolve with the same parameters. As a byproduct, in the synchronized phase the state variables of the slave and that of the master are the same, thus allowing to perform measurements impossible on the real system. We review some aspects of master-slave synchronization using a subset of variables, with intermittent coupling. We show how synchronization can be achieved when some of the state variables are available for direct measurement using a simulated annealing approach, and also when they are accessible only through a scalar function, using a pruned-enriching ensemble approach, similar to genetic algorithms without cross-over.

Keywords: Master-slave synchronization, data assimilation, parameter estimation, Lorenz system

1. Introduction

There are many cases in which one is interested in forecasting the behaviour of a chaotic system, the emblematic one is meteorology. The main obstacle is, of course, that in chaotic systems by definition a small uncertainty amplify exponentially in time [1]. Moreover, even if one assumes to have a good computational model of a natural, chaotic system, the exact value of parameters is needed to have a faithful representation of dynamics.

Schematically, one can assume that the natural, or target system is well represented by some dynamical system, possibly with noise. One can measure some quantities on this system, but in general one is not free to choose which variable (or combination of variables) to measure nor to perform measurements at any rate.

On the other hand, if one has a good knowledge of the system under investigation, i.e., it can be modeled with good accuracy, a simulated “replica” of the system can be implemented on a computer. If moreover one has the possibility of knowing the parameters and the initial state of the original system, so to keep the replica synchronized to it (when running at the same speed), then the replica can be used to perform measurements otherwise impossible, and to get accurate forecasting (when run at larger speed).

In general, the problem of extracting the parameters of a system from a time-series of measurement on it, is called data assimilation [2].

The problem in data assimilation is that of determining the state of the system and the parameters by minimizing an error function between data measured on the target and the respective data from the simulated system. A similar task is performed by the back-propagation techniques in machine learning [3,4].

The goal of this paper is that of approaching this problem from the point of view of dynamical systems [5]. In this context the theme of synchronization is quite explored [6], but it has to be extended in order to cover the application field of data assimilation.

The synchronization technique is particularly recommendable when the noise (in the measure and intrinsic to the system) is small, i.e., when the target system can be assumed to be well represented by a deterministic dynamics, even if this dynamics is not continuous (as in Cellular Automata [7]).

We shall investigate here the application of synchronization methods to the classic Lorenz system [8].

We shall start from the classic Pecora-Carrol master-slave synchronization scheme, recalled in Section 2, in which the value of some of the state variables of the master is imposed to the corresponding variables of the slave system. This choice of coupling variables is denoted “coupling direction” in the tangent space of the system.

The synchronization threshold is connected to the value of the conditional Lyapunov exponent [9,10], i.e., the exponential growing rate along the difference space between master and slave, as reported in Section 3.

This scheme will be then extended to partial synchronization, i.e., to the case in which only a portion of the value of the state variables of the master system signal is fed to the slave, as shown in Section 4. We can thus establish the minimum fraction of signal (coupling strength) able to synchronize the two replicas, that depends on the coupling direction.

However, one cannot pretend to be able to perform measurements in continuous time, as done in the original synchronization scheme, in which the experimental reference system was a chaotic electronic circuit.

Therefore, we dealt with the problem of intermittency, i.e., performing measurements, and consequently applying the synchronization scheme, only at certain time intervals. We show that the synchronization scheme works also for intermittent measurements, provided that the coupling strength is increased, as shown in Section 5.

In the case of systems with different parameters, the synchronization cannot be complete, and we speak of generalized synchronization. We shall show in Section 7 that the distance among systems can be interpreted as measure of the error, and exploited to obtain the “true” value of parameters, using a simulated annealing scheme.

Finally, it may happen that the variables of the systems are not individually accessible to measurements, a situation which forbids the application of the original method. In this case one can still exploit schemes inspired by statistical mechanics, like the pruned and enriching one, simulating an ensemble of systems with different state variables and parameters, and selecting the instances with less error, cloning them with perturbations. This kind of genetic algorithms is able to approximate the actual values of parameters, as reported in Section 8.

Conclusions are drawn in the last section.

2. Pecora-Carrol synchronization

In 1990 Louis Pecora and Thomas Carrol introduced the idea of master-slave synchronization [9]. They studied simulated systems, like the Lorenz [8] and Rössler [11] ones, and also experimental electronic circuits [12].

They considered two replicas of the same system, with same parameters. One of the replica (the master) was left evolving unperturbed. Some of state variables of the master replaced the corresponding variable of the slave system, which is therefore no more autonomous. This topic was further explored in a 2015 paper [10].

In order to get the main idea of this master-slave (or replica) synchronization, let us consider first a one-dimensional, time-discrete dynamical system, like for instance the logistic map [13]. It is defined by an equation of the kind

$$x_{n+1} = f(x_n), \quad (1)$$

where n labels the time interval. The usual Lyapunov exponent λ is defined as

$$\lambda = \lim_{N \rightarrow \infty} \sum_{n=1}^N \log(f'(x_n)), \quad (2)$$

where $f'(x) = df/dx$. Let us now introduce a replica X such that

$$\begin{aligned} x_{n+1} &= f(x_n); \\ X_{n+1} &= (1-p)f(X_n) + pf(x_n), \end{aligned} \quad (3)$$

where p is the coupling parameter (coupling strength).

For $p = 0$ the two maps are uncoupled, and since they are assumed to be chaotic, they generally take different values. For $p = 1$ map X immediately synchronizes with map x . There is a critical value p_c of the coupling parameter for which the synchronized state is stable, and it is related to the Lyapunov exponent λ ,

$$p_c = 1 - \exp(-\lambda), \quad (4)$$

as can be seen by taking the linearized difference among the maps.

This scenario holds also for higher-dimensional (K) maps, which can be written as

$$\mathbf{r}_{n+1} = \mathbf{F}(\mathbf{r}_n), \quad (5)$$

where \mathbf{r}_n denotes the state of the system at iteration n and has K components. The system has now K Lyapunov exponents, of which at least one, λ_{MAX} , is assumed to be positive.

The synchronization procedure in this case is

$$\begin{aligned} \mathbf{r}_{n+1} &= \mathbf{F}(\mathbf{r}_n); \\ \mathbf{R}_{n+1} &= (\mathbb{I} - pC)\mathbf{F}(\mathbf{R}_n) + pC\mathbf{F}(\mathbf{r}_n), \end{aligned} \quad (6)$$

where now the coupling is implemented by means of a diagonal matrix C , with diagonal entries equal to zero or one, defining the coupling directions. In the following we shall indicate the diagonal of C as $\mathbf{C} = [c_x, c_y, c_z]$.

In the case in which $C = \mathbb{I}$, i.e., all entries are mixed with the same proportion, the stability of the synchronized phase is again related to the maximum Lyapunov exponent λ_{MAX} ,

$$p_c = 1 - \exp(-\lambda_{\text{MAX}}), \quad (7)$$

since the evolution of an infinitesimal difference among the replicas, δ , is given by

$$\delta_{n+1} = (1-p)J(\mathbf{x})\delta_n, \quad (8)$$

where J is the Jacobian of F

$$J_{ij} = \left(\frac{\partial F_i}{\partial x_j} \right)_{x=x_n}. \quad (9)$$

This scheme can be extended to continuous-time systems, by replacing the entries from one system in the coupled differential equation, so

$$\begin{aligned} \dot{\mathbf{r}} &= \mathbf{F}(\mathbf{r}); \\ \dot{\mathbf{R}} &= (\mathbb{I} - pC)\mathbf{F}(\mathbf{R}) + pC\mathbf{F}(\mathbf{r}). \end{aligned} \quad (10)$$

where now the Lyapunov exponents of the original system are given by the eigenvalues of the (symmetrized) time-average of Jacobian Λ

$$\Lambda = \lim_{t \rightarrow \infty} \frac{1}{2t} \left(\int_0^t J(\mathbf{r}(t)) + \left(\int_0^t J(\mathbf{r}(t)) \right)^T \right). \quad (11)$$

In practice, however, the differential equation Eq. (10) is implemented as a map, by discretizing the time, and therefore Lyapunov exponents are computed as in the previous case. Using a simple Euler scheme, we have $t = n\Delta t$ and $\mathbf{r}(t) = \mathbf{r}(n\Delta t) = \mathbf{r}_n$, and

$$\mathbf{r}_{n+1} = \mathbf{r}_n + \mathbf{F}(\mathbf{r}_n)\Delta t. \quad (12)$$

In the tangent space

$$\delta\mathbf{r}_{n+1} = (1 + J(\mathbf{r}_n)\Delta t)\delta\mathbf{r}_n, \quad (13)$$

and the maximum Lyapunov exponent λ_{MAX} is

$$\lambda_{\text{MAX}} = \lim_{n \rightarrow \infty} \frac{1}{n\Delta t} \sum_n \log(1 + J(\mathbf{r}_n)\Delta t) \simeq \lim_{n \rightarrow \infty} \frac{1}{n} \sum_n J(\mathbf{r}_n). \quad (14)$$

The average growth of the distance $\delta\mathbf{r}$ is

$$\delta\mathbf{r}(t) = \delta_0 \exp(\lambda_{\text{MAX}} n\Delta t) = \exp(\lambda_{\text{MAX}} t), \quad (15)$$

for time intervals such that the linearized approximation is valid.

Notice that in the continuous-time version the Jacobian and the Lyapunov exponents are defined in units of the inverse of time.

3. Conditional coupling

In the original Pecora-Carrol implementation [10], the signal from the master is fully extracted, i.e., $p = 1$, so that for the \mathbf{R} system, either a component is untouched, or it is derived from the replica (\mathbf{r}).

To be explicit, for the Lorenz system we have

$$\begin{aligned} \dot{x} &= \sigma(y - x), \\ \dot{y} &= -xz + \rho x - y, \\ \dot{z} &= xy - \beta z. \end{aligned} \quad (16)$$

For instance, for a full coupling along the x direction ($\mathbf{C} = [1, 0, 0]$), the replica will follow the equation

$$\begin{aligned} X &= x, \\ \dot{Y} &= -xZ + \rho x - Y, \\ \dot{Z} &= xY - \beta Z. \end{aligned} \quad (17)$$

It is possible to define the sub-Lyapunov [9] or, better, the *conditional* Lyapunov exponents [10], by iterating the equation for the difference $\delta\mathbf{r}$ in the tangent space. For instance, for the Lorenz system coupled along the x direction ($\mathbf{C} = [1, 0, 0]$), we have

$$\frac{d}{dt} \begin{pmatrix} \delta y \\ \delta z \end{pmatrix} = \begin{pmatrix} -1 & -x \\ x & -\beta \end{pmatrix} \cdot \begin{pmatrix} \delta y \\ \delta z \end{pmatrix}, \quad (18)$$

giving two conditional exponents. The system synchronizes if both of them are negative.

Fig. 1 we report the value of the maximum conditional Lyapunov exponent as a function of the coupling strength p for the $\mathbf{C}_x = [1, 0, 0]$, $\mathbf{C}_y = [0, 1, 0]$ and $\mathbf{C}_z = [0, 0, 1]$ coupling directions.

As noted also in Ref [10], the Lorenz system synchronizes if coupled along the x and y directions, but not along the z one, even for $p = 1$.

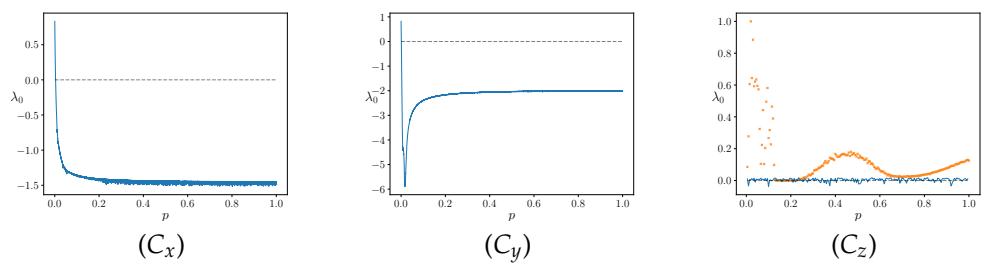


Figure 1. Conditional Lyapunov maximal exponent for different coupling directions. $C_x = [1, 0, 0]$, $C_y = [0, 1, 0]$, $C_z = [0, 0, 1]$. In the last one, in orange, we also show the distance d (normalized at its maximum value obtained in the simulation) between the master and slave state variables for different coupling strengths (see also Fig. 2).

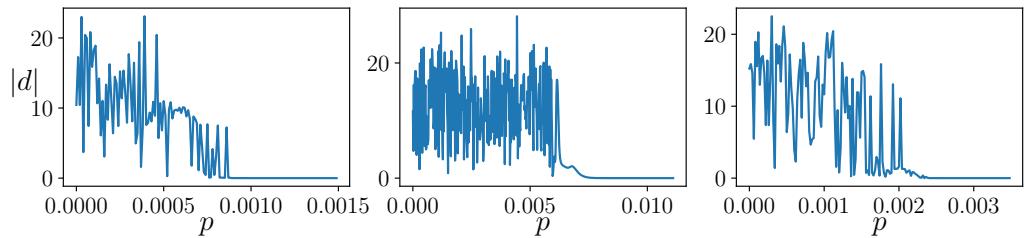


Figure 2. Asymptotic distance d as function of the coupling strength p for different coupling directions. From the left to the right: $C = [1, 1, 1]$, $C_x = [1, 0, 0]$, $C_y = [0, 1, 0]$.

Our observable was the average distance d ,

$$d(t) = \|\mathbf{r}(t) - \mathbf{R}(t)\| = \sqrt{(x(t) - X(t))^2 + (y(t) - Y(t))^2 + (z(t) - Z(t))^2},$$

$$d = \frac{1}{T} \int_0^T d(t) dt, \quad (19)$$

computed after a proper transient.

In Fig. 2 we show d as a function of the coupling strength p for different coupling directions. For all the simulations we set $\sigma = 10$, $\beta = 8/3$, $\rho = 28$, and we use the Euler integration scheme to integrate the equation with temporal step $dt = 10^{-3}$. After a transient time of free evolution we couple the master and the slave system every time step. At $T = 100$ (simulation end time) we compute the distance d between the two systems.

4. Partial conditional coupling

We can naturally generalize the definition of maximum conditional Lyapunov exponents also when we do not have a full coupling, i.e., $p < 1$, looking for the synchronization threshold p_c , for which $d(p)$ goes to zero.

From the plots reported in Fig. 2, one can see that for the x and y coupling there is a well defined synchronization threshold, similar to that obtained for the uniform coupling. For the z coupling, the maximum Lyapunov exponent is always positive (albeit small), and no synchronization is possible.

This behaviour is showed in Fig. 1, where we plot the maximum conditional Lyapunov exponent for different coupling strength p and different coupling direction C . For the z coupling, we also plot the distance $d(p)$ for some p (as Fig. 2).

One can see also that in the un-synchronized phase, the distance $d(p)$ exhibits a non-monotonous behavior, except in the vicinity of the synchronization transitions, Fig. 2. This aspect will be analyzed in Section 7.

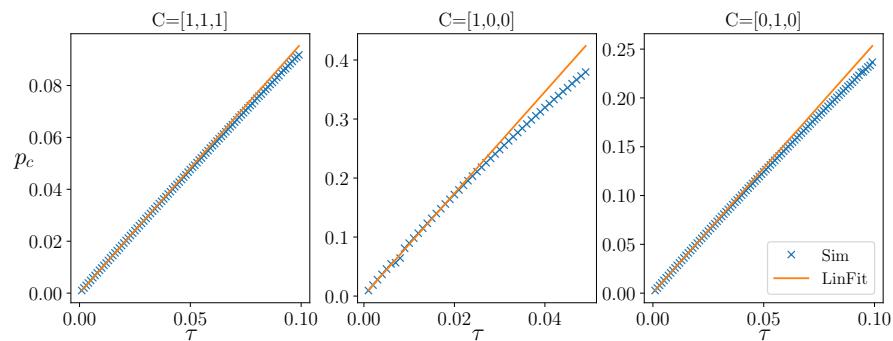


Figure 3. The dependence of the synchronization threshold on the intermittent parameter τ such that $\tau = k\Delta t$ for different coupling directions and it's linear fit obtained using the first 20 time steps. The other parameters of the simulation are: $dt = 10^{-3}$ and $k_{max} = 50$ for $C = [1, 0, 0]$, $k_{max} = 100$ for the others.

	$C = [1, 1, 1]$	$C_x = [1, 0, 0]$	$C_y = [0, 1, 0]$
λ_{lin}	0.995	8.840	2.670
λ_{fit}	0.961	8.619	2.555

Table 1. Maximum conditional Lyapunov exponent estimated using the asymptotic distance $|d|$ between Master and coupled Slave systems (λ_{sim}) and derived from linear fit (λ_{fit}).

5. Intermittent synchronization

In real applications, it is generally impossible to get a signal from one system and inject it into the replica in a time-continuous way. Pecora and Carroll were able to do it using electronic circuits [9], but if one need to pass through a measurement system, it is expects that this device has a finite bandwidth, i.e., a finite measurement time.

So the question becomes: is it possible to synchronize two replicas measuring one quantity only at time intervals τ ?

Numerically, if the equations are integrated using a constant time step Δt , this means that the replacement or mixing of components is applied every k time steps, so that $\tau = k\Delta t$.

For homogeneous coupling ($C = [1, 1, 1]$), the linear analysis for simply tells that (Equation 7)

$$p_c = 1 - \exp(-\lambda_{MAX}\Delta t \cdot k) = 1 - \exp(-\lambda_{MAX}\tau), \quad (20)$$

i.e., intermittent synchronization is equivalent to the standard synchronization for a system with a larger Lyapunov exponent.

Then, for small τ , we have

$$p_c = \lambda_{MAX}\tau. \quad (21)$$

This relationship holds numerically also for other coupling directions, as shown in Fig. 3. In the figure we also plot the linear fit obtained using the first 20 points.

The estimated values obtained from the linear fit (λ_{fit}), compared with the values (λ_{lin}) computed using equation 21 with p_c estimated numerically (Fig. 2), are summarized in Table 1.

6. Generalized synchronization

What happens if the two coupled replicas evolve using different values of parameters σ, ρ, β [6,14]? Even when the coupling parameters (directions and intensity) are above threshold, the distance d remains finite. Let us indicate with σ', ρ', β' the parameters of the slave system.

We can define a parameter distance D ,

$$D = \sqrt{(\sigma - \sigma')^2 + (\rho - \rho')^2 + (\beta - \beta')^2}, \quad (22)$$

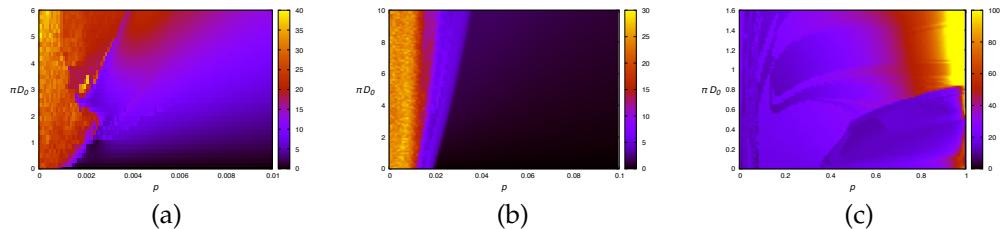


Figure 4. Heat map of the state-variable distance d for different values of parameter coupling $D = \pi D_0$ ($0 \leq \pi \leq 1$) and state-variable coupling p for some state-variable coupling directions C and parameter coupling direction χ . (a) $C = [1, 1, 1]$, $\chi = [1, 1, 1]$; (b) $C_x = [1, 0, 0]$, $\chi = [1, 1, 1]$; (c) $C_z = [0, 0, 1]$, $\chi = [1, 1, 1]$. The line $\pi = 0$ corresponds to Fig. 2

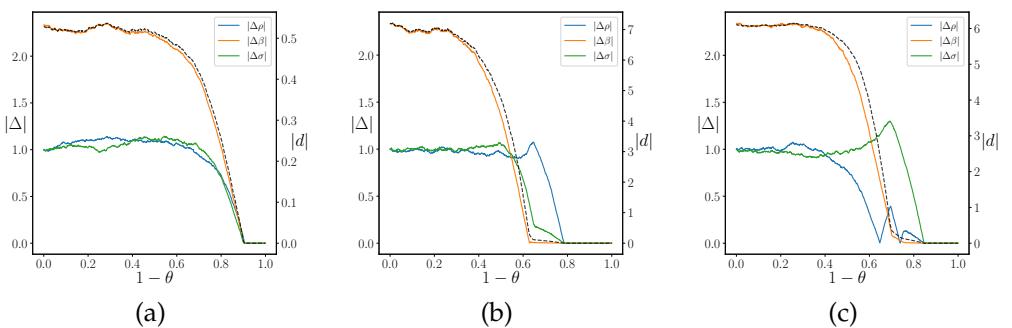


Figure 5. The state-variable distance d (dashed line) and parameter distance (color) as a function of temperature θ for $p = 0.6 \gg p_c$ and different coupling directions C . (a) $C = [1, 1, 1]$; (b) $C = [1, 0, 0]$; (c) $C = [0, 1, 0]$. We set $\theta = 1$, $\epsilon = 10^{-4}$, $T = 100$.

which is zero for the previous coupling schemes. We can also generalize the coupling among replicas, similar to what done for state variables, Eq.(6), introducing a parameter coupling direction $\chi = [\chi_\sigma, \chi_\rho, \chi_\beta]$ and a strength π so that the parameters of the slave replica are

$$\begin{cases} \sigma' = \sigma + \chi_\sigma \pi (\sigma_1 - \sigma), \\ \rho' = \rho + \chi_\rho \pi (\rho_1 - \rho), \\ \beta' = \beta + \chi_\beta \pi (\beta_1 - \beta), \end{cases}$$

where $\sigma_1, \rho_1, \beta_1$ are the values of parameters corresponding to $\pi = 1$, reachable according to the “direction” χ .

We can notice that the distance among state variables d decreases smoothly with $p - p_c$ and D only for a small interval of D near zero. Clearly, $d > 0$ for $p < p_c$, even for $D = 0$, which is what we have seen in Section 4.

Some simulation results are presented in Fig. 4, in which the asymptotic state-variable distance d is reported as a function of the distance between parameters $D = \pi D_0$ and state-variable coupling p . The parameter coupling π always goes from 0 to 1, so the initial parameter distance D_0 corresponds to the larger value of D . The line $D = 0$ corresponds to the distance reported in Fig. 5.

Clearly, in the absence of synchronization for $D = 0$ ($C = [0, 0, 1]$, Fig. 5-(c)), there is no synchronization for $D > 0$. In the other cases, there is a region near the synchronized phase in which d goes smoothly to zero. Notice that in Fig. 5-(c) the larger distance d occurs for $p = 1$ and large D . This is probably due to the fact that when coupled along the z directions, the two replicas may stay on different “leaves” of the attractor, which is almost perpendicular to z .

Notice also that the d landscape is not smooth far from the synchronized phase. We consider this aspect in the following section.

7. Parameter estimation

We are now able to exploit the fact the the distance d goes to zero if $p > p_c$ and $D = 0$, thus allowing to determine the parameters of the master system \mathbf{r} by varying the parameters of the simulated replica \mathbf{R} .

However, since the convergence of d to zero is not monotonous with D , we relay of a simulated annealing technique [15], that allows to overcome the presence of local minima. We introduce a fictitious temperature θ , and assume that the distance d is the analogous to an energy to be minimized.

We assume that the the synchronization time τ and the coupling direction C , cannot be modified at will by experimenters, being fixed by the characteristic of the measure instruments and of the actuators. We insure to be in the condition $p > p_c$, i.e., such that if the parameter distance is null, $D = 0$, synchronization occurs.

The idea is the following: we simulate the coupled system for a time interval T , measuring the state variable distance d , after which one of the parameter σ, ρ, β is varied by a small amount. We repeat the simulation starting from the same conditions, except the varied parameter, and compute again d . If d decreases the variation is accepted. It is also accepted if d increases, with a probability

$$p_{\text{acc}} = \exp\left(-\frac{\Delta d}{\theta}\right), \quad (23)$$

otherwise it is discarded.

The temperature θ is slowly lowered (multiplying θ by a factor $1 - \epsilon$) every T time interval. As shown in Fig. 5, in this way it is possible to exploit the synchronization procedure to get the value of the parameters in the master replica. In fact, for θ sufficiently low, the distance $|d|$ between master and slave state variables (dashed line in figure) drops to zero. For similar values of temperature (but not always the same), also the distances $|\Delta|$ between master and slave parameters drop to zero (continuous color lines). Clearly, this procedure works only for deterministic dynamical systems with little or no noise on measurement, and with very low dimensionality.

8. Pruned-enriching approach

The previous scheme cannot be always followed, since we assumed to be able to measure some of the variables of the master system and inject this value (at least intermittently) in the slave one.

However, this might be impossible, either because the state variables x, y, z are not accessible individually, or because those that are accessible do not allow the synchronization (for instance, if only the z variable is accessible, as illustrated in Section 4).

We can take profit of the Takens theorem [16], that states that the essential features (among which the maximum Lyapunov exponent) of a strange attractor $\{\mathbf{r}(t)\}_{t \rightarrow \infty}$ can be reconstructed by intermittent observations $w_n = f(\mathbf{r}(n\Delta T))$ using the time series $w_n, w_{n-1}, w_{n-2}, \dots$ as surrogate data, provided that their number is larger than the dimensionality of the attractor [17]. Other conditions are that the observation interval ΔT be large enough to have observations w_n sufficiently spaced, but not so large that the w_n are scattered along the attractor, making the reconstruction of the trajectory impossible. It is therefore generally convenient to take an interval Δt substantially larger than the minimum $\Delta t = \tau$, but of the same order of magnitude.

What is interesting, is that one can choose for f an arbitrary function of the original variables, provided that their correspondence is smooth. We therefore employed a method inspired by the pruned-enriching technique [18,19] or genetic algorithm without crossover [20].

We assume that we can only measure a function $f(x, y, z) = f(\mathbf{r})$ of the master system, at time intervals τ . The master system evolves with parameters $\mathbf{q} = (\sigma, \rho, \beta)$.

We simulate an ensemble $\{\mathbf{R}_i\}_{i=1,\dots,h}$ composed by h replicas, each one starting from state variables $\mathbf{R}_i(0)$ ($i = 1, \dots, h$) and evolving with the same equation as the master

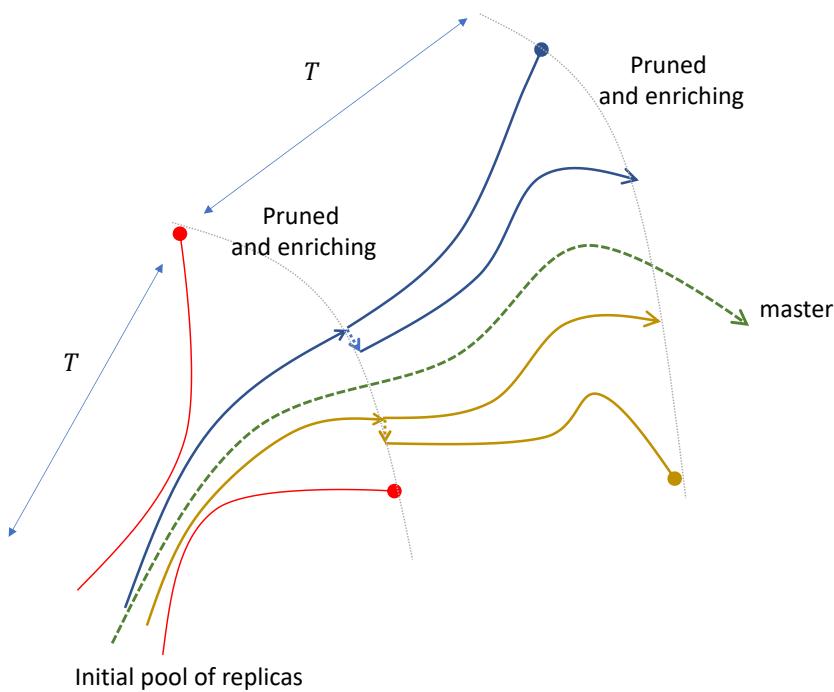


Figure 6. The schematic of the pruned-enriching method. The variation of the duplication of the nearer replicas is either on one of the state variables or one of parameters.

one with parameters $\{Q_i\}_{i=1,\dots,h}$, where $Q_i = (\sigma_i, \rho_i, \beta_i)$. At beginning $R_i(0)$ and Q_i are random quantities.

We let the ensemble evolve for a time interval T and then we measure the distance $d_i = |f(\mathbf{r}(T)) - f(\mathbf{R}_i(T))|$. We sort the replicas according with d_i and we replace the half with larger distances following an evolutionary law based on a cloning and perturbation scheme, as shown in Fig. 6. For a more detailed description of the procedure we can refer to algorithms 1 and 2.

First of all we need to set the initial condition. Given a set of measures $\{f(\mathbf{r}(t))\} = \{f(\mathbf{r}_n)\}$, for $t = t_0, t_0 + \tau, \dots, t_0 + n\tau, \dots, T$, we map the dynamics in an embedding space of size d_e defined by the vectors $\mathbf{w}_n = (f(\mathbf{r}_n), f(\mathbf{r}_{n-1}), \dots, f(\mathbf{r}_{n-d_e+1}))$. Then we randomly initialize the initial conditions $\mathbf{R}_i(0)$ and the initial value of the parameters Q_i of the h replicas, in a range consistent with the physical values. To create the initial embedding vectors of the replicas, we evolve the ensemble for a time $T_{Trans} \geq d_e\tau$.

Now we can start with the optimization problem. For a number of repetitions M , we evolve the ensemble using a time step dt up to time $t = T$. At each further interval τ we update the embedding vector w_i substituting the oldest measurement with the new one and compute the euclidean distances d_i between the W_i and the reference $w(t)$ computed on the master system.

The distance d_i is used as cost function of our optimization problem. In the *Parameters updating step* of Algorithm 2, we sort the elements of the set in ascending order according to d_i and replace the second half of the set with a copy of the first half with either a random perturbation of amplitude δ in one of the parameters, or a random perturbation of the state variables, see Algorithm 2.

We add some check to be sure to not have inconsistent values of parameters (in our case all the Q_i need to be positive). After that we compute \tilde{Q} , the estimated parameters, as mean of the first half of the ensemble elements.

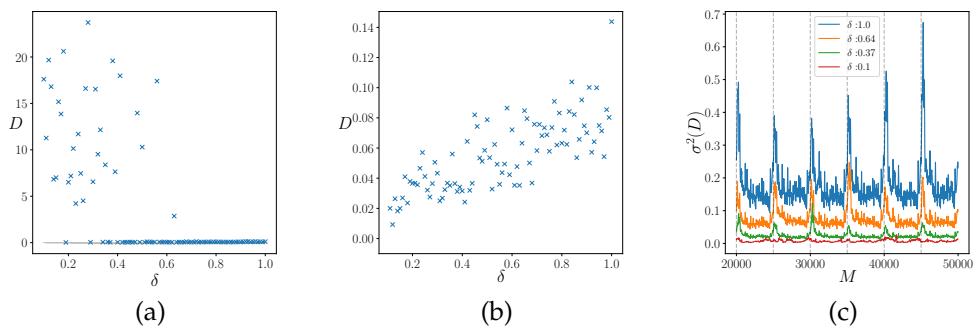


Figure 7. Parameters distance D at interval M for different amplitudes δ with (a) Q_i randomly initialized in the interval $(0.5, 30)$ and (b) Q_i initialized near the “true” values Q by adding a random noise of amplitude $\epsilon = 0.5$. (c) Variance of the distance D for different amplitudes δ for the last interval. The vertical lines indicate when the ensemble was restored to $t = 0$. The Q_i are initialized as in (a).

Since in general we assume $M > T/dt$, the number of time steps in the simulated trajectories, when $t = T$ we restart the ensemble simulation at the initial time step, restoring initial random condition in the state space but without changing the parameters, and we recompute the initial vector W_i evolving the ensemble member up to time $t = T_{Trans}$.

We analyze the convergence problem for different value of δ . We suppose to have access at only the x component of the real system, i.e. our measurement function is simply $f(\mathbf{r}) = x$. We evolve the system up to $T = 100$ with time step $dt = 0.01$ and we measure every $\tau = 0.2$. We embed the system in a embedding space of dimension $d_e = 5$.

The ensemble is composed by $h = 10000$ replicas and we repeat the procedure for $M = 50000$ times. The final results are showed in Fig. 7. In Fig. 7-a we initialize the ensemble parameters randomly in the interval $(0.5, 30)$ and we measure the distance D in the parameters space (Eq. (22)) for different δ . Notice that, starting from a large initial distance, larger values of δ , are more effective for the convergence. The opposite phenomenon is reported in Fig. 7-b. Here we assume to approximately know the true parameter values with an error ϵ and we test the dependence of the amplitude δ in a fine tuning regime. In this case, starting from a relative small distance, smaller values of δ are more effective.

In Fig. 7-c instead we show the behavior of the variance of the parameter distance D during the optimization for different amplitude δ . With a small amplitude δ for the parameters updating step, the elements of the ensemble rapidly converge to the local minimum before having explored the parameter space sufficiently so, small amplitude values of δ can be useful only for a fine tuning approach. Using large values of δ instead is helpful to better explore the parameters landscape and allows to converge to the true values but with noisy results. These considerations suggest the use of an adaptive value of the parameter δ .

Inspired from these results we modify the updating ruled introducing a variable amplitude $\delta = (\delta_1, \delta_2, \dots, \delta_R)$, where, for every ensemble members i , $\delta_i = \delta_i(d_i)$. Then, for each replica we defined an amplitude that modulates the variation step in an self-consistent manner. To test this choice we simply put

$$\delta_i = d_i. \quad (24)$$

In Fig. 8 we plot the behaviour of 50 ensemble members randomly chosen in the pruned-enriching optimization problem with the amplitude factor δ defined as in Eq. (24). We suppose to be in the same situation of the last simulations, so we measure the real system only in the C_x direction every $\tau = 0.2$. The ensemble dimension is equal to $h = 10000$

	real	ens. mean	ens. variance
σ	10.000000	9.997584	0.003873
β	2.666667	2.664959	0.001000
ρ	28.000000	28.017719	0.004583

Table 2. Estimated parameters obtained using the pruned-enriching algorithm with δ variable. On the last column we also show the variance on the ensemble members of the estimated parameters.

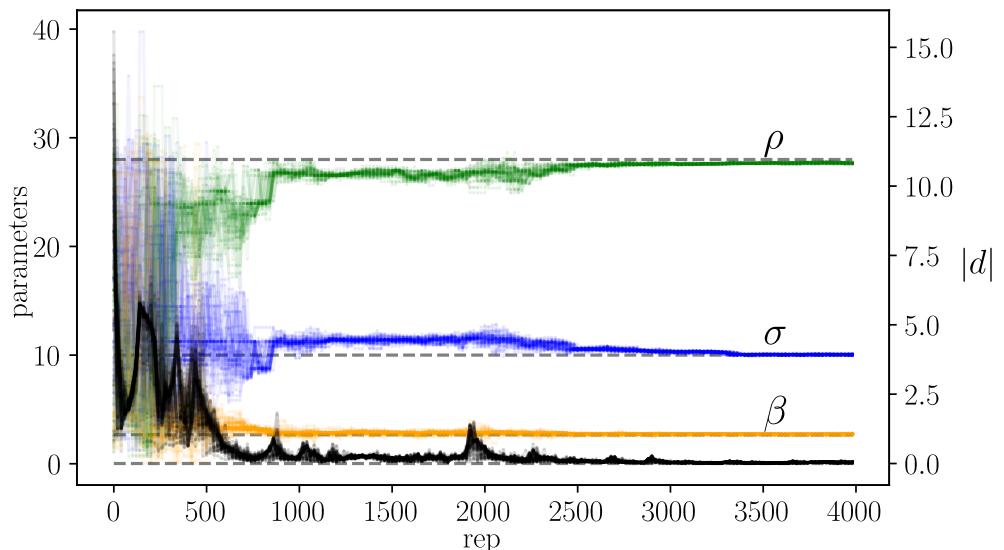


Figure 8. The distance between variable (black points, right axes) and parameters (left axes) as a function of time in the pruned-enriching method for 50 randomly selected replicas of the $h = 10000$ used to estimated the parameters. We consider the situation where only measurement in coupling direction $C = [1, 0, 0]$ are available at every $k = 20$ temporal steps dt , so we embed our time series in an embedding space of dimension $d_e = 5$.

and we choose the embedding dimension $d_e = 5$. In this simulation we run the algorithm for $M = 80000$ intervals. The numerical results as show in Table 2.

The pruned-enriching procedure is similar to a genetic algorithm without cross-over. In general the cross-over operation aims at combining locally-optimised solutions, and depends crucially by the coding of the solution in the “genome”. When the cost function is essentially a sum of functions of variables coded in the solution in nearby zones, the cross-over can indeed allow a jump to an almost-optimal instance. In this case we encode the parameters simply using their current values (a single genome is a vector of 3 real numbers), so there is no indication for this option to be present. It is possible however to pick parameters from “parents” instead of randomly altering them. Since the pool of tentative solutions is obtained by cloning those corresponding to the lowest distances from the master one, we expect little improvements using parameter exchange.

To add the cross-over in our procedure we modify the algorithms such that, for every element of the second half of the ensemble, we chose randomly if update the parameters as Algorithm 2 or do the cross-over step generating the new replica from two parents randomly chosen from the first half of the ensemble. Children inherit randomly two of the parameters from one parent and one from the other.

Without any a priori information about the true parameter values we randomly initialize the initial states and the parameters, so, using the cross-over operation can introduce a bias in the early stages of the optimization problem, as can be seen in Fig. 9, where we compare the estimated parameters for different repetitions using an amplitude

Algorithm 1 Pruned-enriching algorithm.

```

Require:  $\{w(t)\}, \delta, Q_i, R_i, dt, \tau, M, T_{Trans}$ 
 $m \leftarrow 0$ 
 $t \leftarrow 0$ 
while  $m < M$  do
     $m \leftarrow m + 1$ 
    if  $t == T$  then
         $t \leftarrow 0$ 
         $W \leftarrow$  Ensemble evolution up to  $t = T_{Trans}$  and store measure every  $\tau$ 
    else
        if  $t \propto \tau$  then
             $W \leftarrow$  Update with  $f(R)$ 
             $d \leftarrow$  Euclidean distance  $d(w(t), W)$ 
             $Q \leftarrow$  Parameters updating step
             $\tilde{Q} \leftarrow$  Mean of first  $R/2$  ensemble elements ▷ Algorithm 2
        end if
         $R \leftarrow$  Evolution step with time step  $dt$  and  $Q$  as parameters
         $t \leftarrow t + dt$ 
    end if
end while
return  $\tilde{Q}$ 

```

Algorithm 2 Parameters updating step.

```

Require:  $R, \delta, d, W, w(t)$ 
 $\text{index} \leftarrow \text{argsort}(d)$  ▷ return index of d sorted in ascending order
for  $i$  in range( $R/2, R$ ) do
     $Q_i \leftarrow Q_j$  with  $j$  random integer in  $(0, R/2)$ 
    if  $\text{random}(0, 1) < 0.5$  then
         $k \leftarrow$  random integer in  $(1, 2, 3)$ 
         $Q_{ik} \leftarrow Q_{jk} + \text{random}(-\delta, \delta)$ 
        while  $Q_{ik} < 0$  do
             $Q_{ik} \leftarrow Q_{jk} + \text{random}(0, \delta)$ 
        end while
    else
         $R_{ik} \leftarrow R_{jk} + \text{random}(-\delta, \delta)$ 
    end if
end for
return  $Q$ 

```

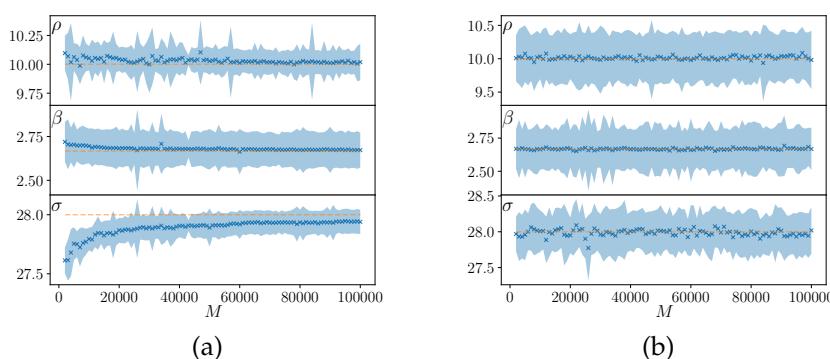


Figure 9. Estimation (x) and standard deviation (blue area) of the parameters computed using the first half of the ensemble with (a) and without (b) cross-over. The true values are also shown (dotted orange lines).

$\delta = 0.8$ with (Fig. 9a) and without (Fig. 9b) cross-over at every $t = T$. As the other simulations we randomly initialize the initial states and parameter values of the ensemble, we evolve the system, after the transient time T_{Trans} , up to $T = 100$ with $dt = 0.01$ and we suppose to be able to measure the x direction with $\tau = 0.2$.

The cross-over allow to make jumps in the parameters space, but in the early stage of the optimization process these jumps can cause the ensemble convergence to wrong values. On the other hand, the cross-over can reduce the variance of the ensemble estimation, so it can help in the last steps, or for fine tuning. Future work is needed to explore more in details this option.

9. Conclusions

We have shown that it is possible to exploit several aspects of the master-slave synchronization to retrieve the parameters of the master system (assumed to be a real, physical one) though a simulated replica. In this way, it is also possible to perform measurements impossible on the real system, using the simulated replica.

We have extended the original Pecora-Carrol synchronization scheme [10], to partial and intermittent coupling.

We have shown that synchronization can be achieved when some of the state variables are available for direct measurement and that the parameters of the original systems can be reconstructed by synchronization using a simulated annealing approach.

We have then shown that the synchronization method can be exploited to retrieve unknown parameters even when variables are accessible only through a scalar function, using a pruned-enriching ensemble approach, similar to genetic algorithms without cross-over, which is then introduced without remarkable improvements.

This works is only a first glimpse into a wide field. The proposed methods can be applied to other dynamical systems, and their limits are still to be precisely defined.

Other important questions concern the dimensionality of systems, since real systems are only exceptionally described by low-dimensional dynamical systems, and the influence of noise that always affect real-life measurements.

Author Contributions: Conceptualization, F.B.; methodology, F.B.; software, F.B. and M.B.; validation, M.B.; formal analysis, F.B.; investigation, F.B. and M.B.; writing—original draft preparation, F.B.; writing—review and editing, F.B. and M.B.; visualization, M.B.; supervision, F.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest. The authors had no role in the collection of data from Wikipedia.

Sample Availability: All codes are available upon request.

References

1. Lorenz, E. *The Essence of Chaos (Jessie and John Danz Lectures)*, paperback ed.; University of Washington Press, 1995; p. 240.
2. Evensen, G.; Vossepoel, F.C.; van Leeuwen, P.J. *Data Assimilation Fundamentals: A Unified Formulation of the State and Parameter Estimation Problem (Springer Textbooks in Earth Sciences, Geography and Environment)*, kindle edition ed.; Springer, 2022; p. 415.
3. Geer, A.J. Learning earth system models from observations: machine learning or data assimilation? *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* **2021**, 379. <https://doi.org/10.1098/rsta.2020.0089>.
4. Bonavita, M.; Alan Geer, P.L.; Massart, S.; Chrust, M. Data assimilation or machine learning? ECMWF Newsletter Number 167 - Spring 2021 <https://www.ecmwf.int/en/newsletter/167/meteorology/data-assimilation-or-machine-learning>, 2021. [Accessed 13-Feb-2023].
5. Strogatz, S.H. *Nonlinear Dynamics And Chaos: With Applications To Physics, Biology, Chemistry And Engineering (Studies in Nonlinearity)*, hardcover ed.; CRC Press, 1994; p. 512.
6. Pikovsky, A.; Rosenblum, M.; Kurths, J. *Cambridge nonlinear science series: Synchronization: A universal concept in nonlinear sciences series number 12*; Cambridge nonlinear science series, Cambridge University Press: Cambridge, England, 2003.
7. Bagnoli, F.; Rechtman, R. Synchronization and maximum Lyapunov exponents of cellular automata. *Physical Review E* **1999**, 59, R1307–R1310. <https://doi.org/10.1103/physreve.59.r1307>.
8. Lorenz, E.N. Deterministic Nonperiodic Flow. *Journal of the Atmospheric Sciences* **1963**, 20, 130–141. [https://doi.org/10.1175/1520-0469\(1963\)020<0130:dnf>2.0.co;2](https://doi.org/10.1175/1520-0469(1963)020<0130:dnf>2.0.co;2).
9. Pecora, L.M.; Carroll, T.L. Synchronization in chaotic systems. *Physical Review Letters* **1990**, 64, 821–824. <https://doi.org/10.1103/physrevlett.64.821>.
10. Pecora, L.M.; Carroll, T.L. Synchronization of chaotic systems. *Chaos: An Interdisciplinary Journal of Nonlinear Science* **2015**, 25, 097611. <https://doi.org/10.1063/1.4917383>.
11. Rössler, O. An equation for continuous chaos. *Physics Letters A* **1976**, 57, 397–398. [https://doi.org/10.1016/0375-9601\(76\)90101-8](https://doi.org/10.1016/0375-9601(76)90101-8).
12. Newcomb, R.; Sathyan, S. An RC op amp chaos generator. *IEEE Transactions on Circuits and Systems* **1983**, 30, 54–56. <https://doi.org/10.1109/tcs.1983.1085277>.
13. May, R.M. Simple mathematical models with very complicated dynamics. *Nature* **1976**, 261, 459–467. <https://doi.org/10.1038/261459a0>.
14. Rulkov, N.F.; Sushchik, M.M.; Tsimring, L.S.; Abarbanel, H.D.I. Generalized synchronization of chaos in directionally coupled chaotic systems. *Physical Review E* **1995**, 51, 980–994. <https://doi.org/10.1103/physreve.51.980>.
15. Kirkpatrick, S.; Gelatt, C.D.; Vecchi, M.P. Optimization by Simulated Annealing. *Science* **1983**, 220, 671–680. <https://doi.org/10.1126/science.220.4598.671>.
16. Takens, F. Detecting strange attractors in turbulence. In *Lecture Notes in Mathematics*; Springer Berlin Heidelberg, 1981; pp. 366–381. <https://doi.org/10.1007/bfb0091924>.
17. Packard, N.H.; Crutchfield, J.P.; Farmer, J.D.; Shaw, R.S. Geometry from a Time Series. *Physical Review Letters* **1980**, 45, 712–716. <https://doi.org/10.1103/physrevlett.45.712>.
18. Rosenbluth, M.N.; Rosenbluth, A.W. Monte Carlo Calculation of the Average Extension of Molecular Chains. *The Journal of Chemical Physics* **1955**, 23, 356–359. <https://doi.org/10.1063/1.1741967>.
19. Grassberger, P. Pruned-enriched Rosenbluth method: Simulations of θ polymers of chain length up to 1 000 000. *Physical Review E* **1997**, 56, 3682–3693. <https://doi.org/10.1103/physreve.56.3682>.
20. Mitchell, M. *An introduction to genetic algorithms*; MIT Press: London, England, 1996.

Acknowledgements

We acknowledge useful discussions with Sara Beltrami.