Rapid chaotic synchronization by intermittent driving signals

P.G. Vaidya and Sajini Anand

Abstract—**Synchronization of two identical chaotic systems which starts with different initial conditions, by sending a part of state space to other in a continuous fashion is a well established procedure. This paper discusses synchronization by intermittent driving signals from a part of a system to the other system.**

Here we show numerical evidence that if we were to run the second system on its own until the intermittent information about the first is available, and replacing it, synchronization does take place but it takes a longer time. What we show is a method to speed up this procedure even when the intermittent signals are not that frequent. This has potential application in communication, especially in the area of cryptography. Details of procedure and possible application in cryptography are included in Ref. [8].

Keywords— **Chaotic Cryptography, Synchronization, Secure Communication, Super-key.**

I. Introduction

Consider two systems A and B which are represented by an identical set of differential equations, but with different set of initial conditions. They would be observed to diverge from one another in a short time. However Pecora and Caroll [1] showed that in the case of Lorenz equation (shown below), such systems can achieve perfect synchronization if information about one of the states of system A is used to override the corresponding state in system B.

specifically A is represented as

$$\begin{bmatrix} \dot{x}_0 \\ \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} \sigma(x_1 - x_0) \\ \rho x_0 - x_1 - x_0 x_2 \\ x_0 x_1 - \beta x_2 \end{bmatrix} \tag{1}$$

Let the system B be

$$\begin{bmatrix} \dot{y}_0 \\ \dot{y}_1 \\ \dot{y}_2 \end{bmatrix} = \begin{bmatrix} \sigma(y_1 - y_0) \\ \rho y_0 - y_1 - y_0 y_2 \\ y_0 y_1 - \beta y_2 \end{bmatrix} \tag{2}$$

In that case they were using x_0 from A system to override the y_0 of other system. therefore they were only using these two remaining equations, $y_0 = x_0$

$$\dot{y}_1 = \rho x_0 - y_1 + x_0 y_2 \tag{3.1}$$

$$\dot{y}_2 = x_0 y_1 - \beta y_2. \tag{3.2}$$

Eventhough y_1 and y_2 starts at different initial conditions than x_1 and x_1, both the systems synchronize as time goes on [1]. The Pecora and Caroll systems assume that the feedback is continuous that is x_0 continuously replaces y_0 . It is found that after a time duration, $y_1(t) = x_1(t)$ and $y_1(t) = x_1(t)$. The proof of synchronization using Lyapunov function is already published [2]. The synchronization can be speeded up as shown by [3].

The systems A and B are represented by different metaphors. In one metaphor A is called the master and B is called the slave. In communication metaphor, for Cryptography purposes A is called 'Alice' and B is called 'Bob'; where Alice is trying to send the information to Bob and a part of the sending signal is used to get Bob's system synchronized with that of Alice. For the purpose of this paper we use the metaphor of Master and Slave.

In this paper, we use the full equations of the slave system B, except that we have an option to override y_0 by x_0 whenever the information about x_0 is available. Three possible cases arise. One, where the information is available at a very high rate. In which case synchronization proceeds almost as in the case of the continuous override of y_0 by x_0, since virtually there is no difference. The second case is when the sampling rate is little bit slow. In that case we can keep overriding y_0 by x_0 at slower rate and keep synchronizing the system B with A. We have found that the rate of synchronization can be speeded up by the process we describe below. The third and most important case is when the information is not frequent, i.e. when the driving signal is intermittent. Under this condition speeding up of synchronization is necessary and most the paper talk about how this can be done.

II. Numerical demonstration of synchronization by a straightforward substitution of the intermittent signals

As discussed above, intermittent samples of $x_0(t)$ is sent to the system B. Trajectories of both systems were integrated using equations (1) and (2) by the Runge-Kutta procedure. Whenever the signal $x_0(t)$ is available, it replaces the current value of $y_0(t)$. Since the sampling rate is less, the systems eventually synchronize, but often after a long time.

For the numerical simulation the parameters of the systems were $(\sigma = 10.0, \beta = 8/3, \rho = 29.75)$ and the initial conditions of A were $(1.874 \quad 2.056 \quad 19.142)^T$. B starts with its own initial conditions. Here we assume B knows x_0 and has no idea about x_1 and x_2. For simulation, we assumed them to be zero. So the initial conditions of B are $(1.874 \quad 0 \quad 0)^T$

In the first case, trajectories of system A and B are calculated at every 0.0001 second. The driving signal x_0 was sent to B at every 0.1 second; i.e. we sent every 1000 sample. The results of simulation are shown in Fig. 2 and 3. It can be seen from Fig. 3 that it takes at least 5 seconds for the systems to synchronize. Depending on the initial condition of B, it might take lesser or longer time. If the driving signal is sent with a lesser sampling rate, it takes even longer time to synchronize.

It is important to note that the results of this section and the next are closely related to the results in papers [8,9].

III. Rapid Synchronization-Procedure

The ideas of this section are based on the concept of embedding. Let us denote S_0, S_1, S_2 be three consecutive samples of $x_0(t)$ received at time $t = t, t + h, t + 2h$

$$S_0 = x_0(t); S_1 = x_0(t + h); S_2 = x_0(t + 2h).$$

National Institute of Advanced Studies, Indian Institute of Science Campus, Bangalore 560 012, email: pgvaidya@nias.iisc.erent.in

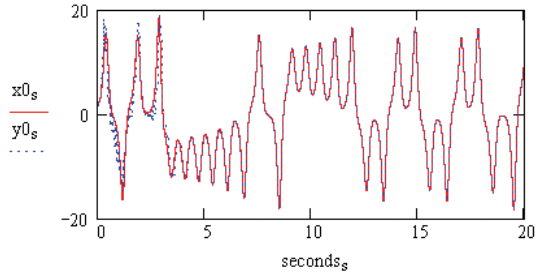

Fig. 1. Trajectories of Master system A and slave system B as a function of time. B eventually synchronizes with A after along time

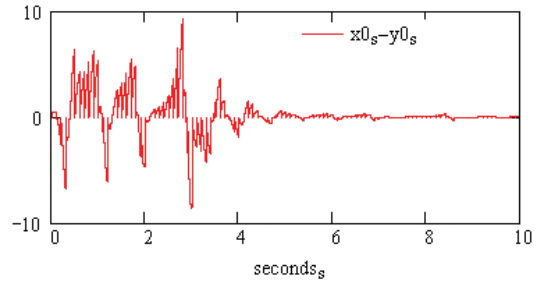

Fig. 2. The difference between the trajectories of A and B plotted against time. B eventually synchronizes with A

The question is to find the initial conditions $x_0(0)$, $x_1(0)$, $x_2(0)$ from the samples of $x_0(t)$ i.e. S_0, S_1, S_2.

Let us reverse this process and say that the initial conditions do determine S_0, S_1, S_2. we know $S_0 = x_0(t)$. For a fixed x_0 let S_1 be a function of x_1, x_2.

$$S_1 = f(x_1, x_2) \tag{4.1}$$

where $f(a, b)$ represent the value of y_0 at time $t + h$ obtained by running a numerical procedure with initial conditions $(x_0 = x_0, \quad x_1 = a, \quad x_1 = b)^T$.

Similarly let

$$S_2 = g(x_1, x_2) \tag{4.2}$$

Our task is to find the zeros of the function,

$$F(x_1, x_2) = \begin{pmatrix} f(x_1, x_2) - S_1 \\ g(x_1, x_2) - S_2 \end{pmatrix} \tag{5}$$

This can be solved by a Vector Newton Raphson-Procedure

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x_1' \\ x_2' \end{bmatrix} - J \begin{bmatrix} x_1' \\ x_2' \end{bmatrix} .F \begin{bmatrix} x_1' \\ x_2' \end{bmatrix} \tag{6}$$

where (x_1', x_2') are the starting guesses and (x_1, x_2) are the improved solutions. J is the Jacobian, and to evaluate J we need at least two auxiliary trajectories near the initial guess of $(x_0, \quad x_1', \quad x_2')^T$.

To get the overall idea behind this, consider Fig. 4. In this concept, we conceive of a map from initial conditions in the usual state space of x_0, x_2, x_2 and a new state space constructed from the samples of x_0, i.e. S_0, S_1, S_2. If we choose some other initial conditions in (x_0, x_1, x_2) space, we would get another point in the S_0, S_1, S_2 space. In fact, in all probabilities, B has some different initial conditions.after receiving the first driving signal sample $y_0(x_0 = y_0)$ of B (and therefore S_0) would agree with A. But x_1, x_2 (or as we have used the notation y_1, y_2) would be different. The S picture describe how B's initial condition conditions can be corrected to synchronize with A

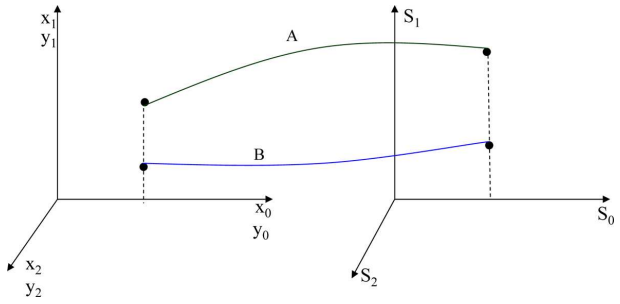

Fig. 3. A sketch of a map between initial conditions and measurements x_0 at $t = 0$, 0.1 and 0.2. Different initial conditions of B and their discrepancies in S space

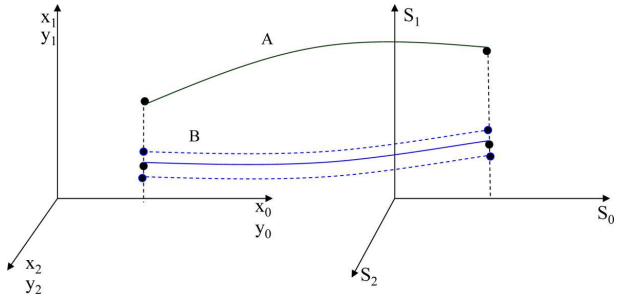

Fig. 4. Map of A's initial conditions and Different guesses for initial conditions of B

Once again the approach of trial and error runs into enormous difficulties. Consider to auxiliary trajectories near the initial guess of B. Based upon these trajectories we can develop a relationship between deviations from B's trajectory in y_1 and y_2 dimension and how they relate to the deviations in the S components at $t = t + h$ and $t + 2h$ respectively. We can see that the errors in the initial trajectories lead to discrepancies in the observations of the component later. If the errors are small, then the discrepancies later would be linearly related to these errors and this can be represented by a matrix. This matrix can be inverted; therefore from errors we can guess back what the initial discrepancies were.

Using the discrepancies occur between the trajectories of A and B on the same inversion matrix, we can improve system B's case. If B's initial condition is fairly close to A's, then the case is improved almost immediately to the exact solution. But if B's initial conditions are farther, several iterations would be needed.

Specially, let us take two additional trajectories which in principle start very close to B's initial conditions. We can calculate the deviations of those new trajectories with the original one, which is essentially the deviations in B's y_0 component. Now

we can derive a relation between those deviations and the initial conditions. We could do this in principle by taking extremely small deviations from B's starting point, using exact equations and finding the transfer matrix. In practice this can be accomplished much better by taking the Jacobian of the non linear equation over a very small region. This scheme is shown in Figure 5.

Returning to Fig. 4, System B has no information of exactly where A starts in x_0, x_1, x_2 space. We can choose $y_0 = x_0$ and some y_1 and y_2. Now, let us runs a simulation using eq. (2) and arrives at a point in the S space. In the S space at $t = 0$, the values of S_0 for both are identical. Let us assume that at the next observation at $t = h$, B's y_0 is $(S_1 - \delta)$ and at the next one at $(t = 2h)$ it is $(S_2 - \mu)$. We would form a column vector using this derivation, called Jump. So,

$$Jump = \begin{bmatrix} S_1 - (y_0)_h \\ S_2 - (y_0)_{2h} \end{bmatrix} = \begin{bmatrix} \delta \\ \mu \end{bmatrix}. \tag{4}$$

Consider one of the additional trajectories of B. Assume that its trajectory is given by

$$[y_0(t) + \eta_0(t) \quad y_1(t) + \eta_1(t) \quad y_2(t) + \eta_2(t)]^T.$$

If η's are very small, the equation for them can be found from the Jacobian of B's equation(2). Thus

$$\frac{d}{dt}\begin{pmatrix} \eta_0 \\ \eta_1 \\ \eta_2 \end{pmatrix} = \begin{bmatrix} -\sigma & \sigma & 0 \\ (\rho - y_2) & -1 & -y_0 \\ y_1 & y_0 & -\beta \end{bmatrix} \begin{pmatrix} \eta_0 \\ \eta_1 \\ \eta_2 \end{pmatrix}. \tag{5}$$

We can solve 3 sets of equations simultaneously. One for the original initial condition of B and two for the additional trajectories we specified, which both follow equation (5).

For both the additional trajectories η_0 is zero. Since we can extend the linearization, we can choose $\eta_1 = 1, \eta_2 = 0$ for one and $\eta_2 = 1$ for the other.

Now, define matrix A as

$$A = \begin{bmatrix} (\eta_0)_h & (\eta_0')_h \\ (\eta_0)_{2h} & (\eta_0')_{2h} \end{bmatrix} \tag{6}$$

where the prime is used for the second trajectory.

The A matrix tells us how unit derivations in x_1 and x_2 initial conditions transform into derivations in S_1 and S_2.

If linearity were to prevail the initial error in B

$$Error = \begin{pmatrix} y_1(0) - x_1(0) \\ y_2(0) - x_2(0) \end{pmatrix} \tag{7}$$

will also get multiplied by the same matrix so that

$$Jump = A \cdot Error. \tag{8}$$

$$Error = A^{-1} \cdot Jumb \tag{9}$$

$$Corrected\ y = \begin{bmatrix} y_0 \\ y_1 + Error_0 \\ y_2 + Error_1 \end{bmatrix}. \tag{10}$$

In general, linearity will not extend to the position of system B. So, following the spirit of Newton–Raphson method, we can choose the corrected value and iterate again. If this procedure does not converge,we need to select an initial condition from a different part of the state space.

IV. Numerical Results

In this section, we would show how synchronization of A and B is possible with only first three samples from A.

We generated A's trajectory using equation (1). We used Runge-Kutta procedure and find x at intervals of 10^{-4} sec. Initial conditions were chosen at $(1.874 \quad 2.056 \quad 19.142)^T$.

In the first simulation, we used $h = 0.1$ sec. Therefore

$$S_0 = x_0(0); \quad S_1 = x_0(0.1); \quad S_2 = x_0(0.2).$$

S_0, S_1 and S_2 are send to B.

We choose several initial conditions for B. One which was quite for away from that of A: $(18.46 \quad 0 \quad 0)^T$. (Of course, $y_0 = x_0 = S_0$)

The iteration procedure worked quite well. Predicted trajectory for system B is shown in Fig. 6. The derivations shown in Fig. 7 proves that the synchronization is almost perfect.

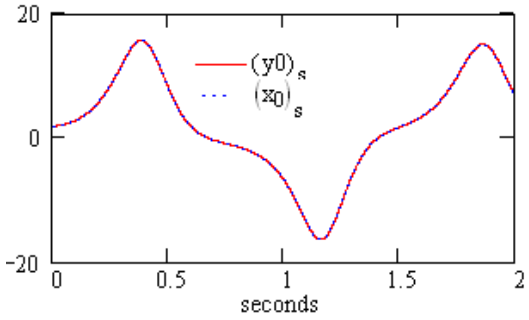

Fig. 5. Evolution of x_0 and predicted y_0 for $h = 0.1$ seconds

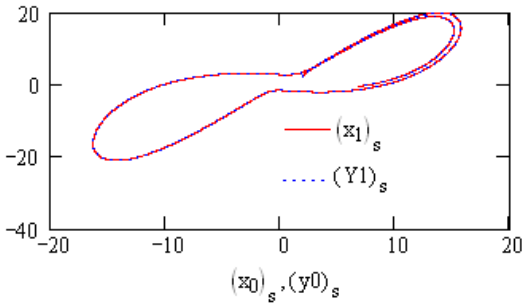

Fig. 6. State space for x and y

Our second simulation used the same initial conditions for A but h was 0.5 seconds, so that

$$S_0 = x_0(0); \quad S_1 = x_0(0.5); \quad S_1 = x_0(1.0).$$

Now if we starts with the an initial guess for B: $(18.46 \quad 0 \quad 0)^T$. it does not converge. However, there is a fairly large neighborhood of initial conditions around A's conditions for which convergence takes place. So, using a strategy akin to simulated annealing, we soon arrive at an initial condition $(18.46 \quad 1 \quad 16.5)^T$.

In this case, the synchronization is once again quite good. This is shown in Figs 9, 10 and 11.

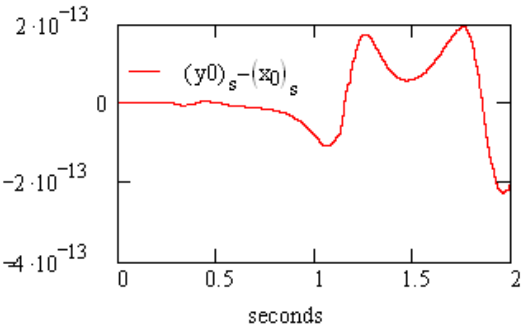

Fig. 7. Difference in x_0 and y_0

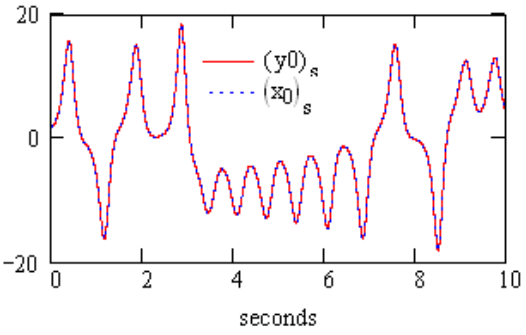

Fig. 8. State space for $h = 0.5$ second

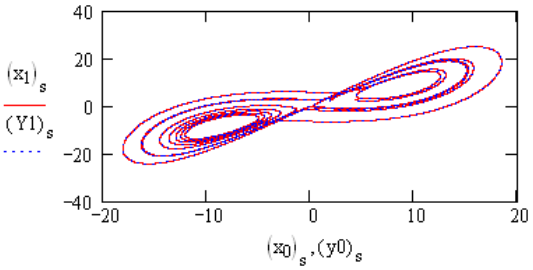

Fig. 9.

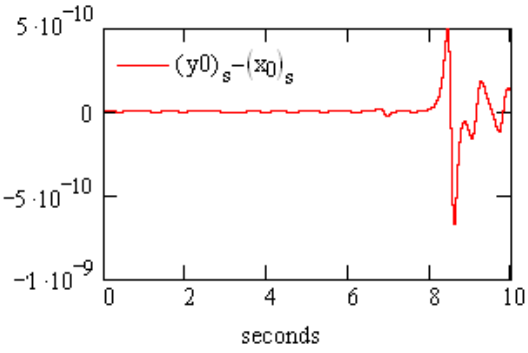

Fig. 10.

V. Conclusions

The theory of chaotic synchronization assumes a continuous feedback from sender to receiver. In practice we send digitally sampled versions, with a fairly high sampling rate. We have shown that a rapid synchronization is possible even with infrequent sampling.This property can find wide applications, for example in the area of cryptography.

Acknowledgements

We would like to thank referee for bringing out our attention to important references.

References

[1] L. M. Pecora and T. L. Carroll, *Phys. Rev. Lett.* **64**, 821 (1990)
[2] Rong He and P. G. Vaidya, Analysis and synthesis of synchronous periodic and chaotic systems, *Phys. Rev.* **A46**, 7387–7392 (1992)
[3] P. G. Vaidya, Monitoring and speeding up chaotic synchronization Chaos, Solitons and Fractals, Volume 17, Number 2, July 2003, pp. 433-439(7)
[4] Rong He and P. G. Vaidya, Implementation of chaotic cryptography with chaotic synchronization, *Phys. Rev.* **E57**, 1532-1535 (1998)
[5] K. M. Cuomo and A. V. Oppenheim, *Phys. Rev. Lett.* **71**, 65 (1993
[6] M. Lakshmanan and S. Rajasekar, Nonlinear Dynamics: Integrability, Chaos, and Patterns, Springer-Verlag, 2003, Ch. 16, India
[7] P.G. Vaidya and Savita Angadi, Decoding chaotic cryptography without access to the super key. *Chaos, Solitons and Fractals* **17(2-3)**, 379-386, (2003)
[8] R.E. Amritkar and Neelima Gupta, Synchronization of chaotic orbits: The effect of a finite time step. *Phys. Rev.* **E47(6)**, 3889-3895 (1993)
[9] Anil Maybhate, R.E. Amritkar and D.R. Kulkarni, Estimation of initial conditions and secure communication. *Intl. J. Bifurcation of Chaos* **13(10)**, 3079-3084 (2003)