Public-channel cryptography using chaos synchronization

Einat Klein[1], Rachel Mislovaty[1], Ido Kanter[1] and Wolfgang Kinzel[2]

[1]*Minerva Center and Department of Physics,*

Bar-Ilan University, Ramat-Gan, 52900 Israel, and

[2]*Institute for Theoretical Physics, University of Würzburg,*

Am Hubland 97074 Würzburg, Germany

Abstract

We present a cryptographic system that is made of two parties with chaotic dynamics that are mutually coupled and undergo a synchronization process, at the end of which they can use their identical dynamical state as an encryption key. The transferred coupling-signals are based non-linearly on time-delayed states of the parties, and therefore they conceal the parties' current state and can be transferred over a *public* channel. To enhance security, the parties are made of chaotic cyclic networks of size N. Synchronization time grows as $logN$, while an attacker trying to synchronize with the parties finds it exponentially difficult with N, due to the linear instability of the chaotic dynamics in high dimension and the structure of the networks.

PACS numbers: 05.45.Vx, 05.45.Gg, 05.45.Xt

The idea of using chaos for creation of secure communication systems has been thoroughly investigated in the past few years and has attracted the attention of many researchers[1–5]. Chaotic systems have the ability to synchronize and they are linearly instable and unpredictable[6], which makes them promising candidates for creating a cryptographic-system. However, until today, chaotic cryptographic-systems always required some "secret parameter" which is known only to the two synchronizing parties: one of the parameters of the system must not be revealed to the attacker, and must be transferred over a private channel. This has always been the weak spot of the chaotic systems known today - the secret parameter in most of the systems is relatively easily found by the error function attack method [8, 9]. In this letter we present a cryptographic-system that creates a *secret* key using a *public* channel, i.e. a key-exchange protocol. It is to our knowledge the first chaos-based cryptographic system generated over a public channel. In our approach, each party uses a chaotic system which is coupled to the other system so that they synchronize. Soon after synchronization one of the chaotic variables is used as an encryption key. Although an attacker, listening to the communication channel, knows all the details of the systems, including the values of the parameters as well as the signals transmitted, he does not manage to synchronize.

The main reason for the advantage of the parties over the attacker is that there is an essential difference between them: the two parties are *mutually* coupled, and are drawn to each other, while the attacker is trying to follow them, and is *one-way* coupled [6]. He is rather "chasing" the two parties while they are moving one towards the other[7]. This advantage of the parties over the attacker is the essence of our cryptographic system.

The first method constructing a secret key over a public channel has been developed in 1976 by Diffie and Hellmann. This method is based on number theory, and it is the basis of almost all modern encryption protocols. Only recently, it has been discovered that a public key exchange protocol can be realized by a physical mechanism as well: two neural networks that are trained by their mutual output bits and synchronize to a common state which is used as a secret key [10].

Here, we consider two chaotic systems that are mutually coupled by a few of their internal variables. This issue has been well studied, but for cryptographic application we have to add two additional ingredients to the chaotic systems: Nonlinearity and time delay of the transmitted signals.

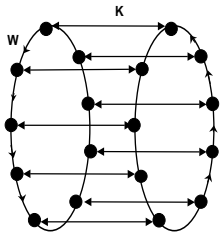

Figure 1: Schematic figure of the two coupled cyclic networks. Each node represents a chaotic system (Lorenz map). Each node is coupled to the preceding node, and to a parallel node in the other network.

The cryptographic system is made of two cyclic networks that are coupled. Each node is coupled to its preceding neighbor in the same network, and to a parallel node in the other network, as shown in Fig. 1. For the dynamics of each node we used the Lorenz system, though other chaotic maps can also be used [6]. During the synchronization process the networks exchange N coupling signals every time step. Eventually they reach an identical state, in which each node is identical to the matching node in the other network, but all the nodes among the same network are de-synchronized. The parties then have a secret key by the size of the networks used, consisting of N degrees of freedom. Hence, this system exhibits synchronization in high-dimensional chaos, and when used for cryptographic purposes, its security increases exponentially with N.

As the coupling signals are transferred publicly, they must be sophisticated enough to hide the state of the systems. We used coupling signals that are based non-linearly on time-delayed values of the systems, and therefore conceal the system's current state, yet still enable synchronization.

An attacker that is tuned to the channel used by the parties, uses the transmitted signals to try to synchronize with them. His probability to synchronize even *partially* with the parties drops exponentially with the size of the network N, although the synchronization time grows like lnN.

Let us now describe the system in more detail. We first consider two networks, 1 and 2, by the size of N, with *linear* time-delayed coupling of their x-value:

$$\begin{aligned}\frac{dx_1^i}{dt} &= 10(y_1^i - x_1^i) + K[x_2^i(t-\tau) - x_1^i(t-\tau)] \\ &\quad + W[x_1^{i-1}(t-\tau) - x_1^i(t-\tau)] \\ \frac{dx_2^i}{dt} &= 10(y_2^i - x_2^i) + K[x_1^i(t-\tau) - x_2^i(t-\tau)] \\ &\quad + W[x_2^{i-1}(t-\tau) - x_2^i(t-\tau)]\end{aligned} \tag{1}$$

$$\begin{aligned}\frac{dy_1^i}{dt} &= 28x_1^i - y_1^i - x_1^i z_1^i \quad & \frac{dy_2^i}{dt} &= 28x_2^i - y_2^i - x_2^i z_2^i \\ \frac{dz_1^i}{dt} &= x_1^i y_1^i - \frac{8}{3} z_1^i & \frac{dz_2^i}{dt} &= x_2^i y_2^i - \frac{8}{3} z_2^i\end{aligned}$$

where K is the coupling strength between the two systems and W is the coupling strength of the inner connections in each system. We first consider the case of just two coupled Lorenz systems ($N = 1$ and $W = 0$).

The ability of two chaotic systems to synchronize when driven by a mutual signal has been subject to vast research [1, 2]. Time delayed coupling has been recently studied [11, 12] and is also observed in biological systems where a time delay occurs between the firing of adjacent neurons. Our numerical investigations showed that with a time delayed-signal, the range of possible K values is limited and synchronization was achieved for a certain range of coupling strength K values only, as described in Fig. 2. The time delay is also limited and synchronization is possible up to $\tau \sim 0.12$ for $K = 8$.

As the dynamics are deterministic, the systems are initialized with random values, otherwise the key would be known.

Using $x(t-\tau)$ as the coupling signal is not secure, therefore we suggest using a nonlinear function of the variable x at previous time steps, $f(t) = f(x(t-\tau_1), x(t-\tau_2))$.

Hence, the first equation of Eqs.1 is replaced by

$$\frac{dx_1}{dt} = 10(y_1 - x_1) + K[f_2(t) - f_1(t)] \tag{2}$$

Which nonlinear function f should be used? As the coupling force is transmitted publicly, it should answer two demands which are somewhat contrast to each other: On one hand, we must choose $f(t)$ that enables synchronization. If we choose something that is too far

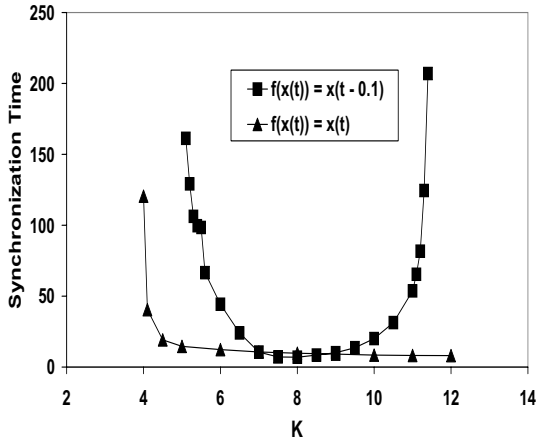

Figure 2: The synchronization time versus K, for the coupling force: $f(x(t)) = x(t)$ (triangles), and the time-delayed coupling force: $f(x(t)) = x(t-\tau)$ (squares),$\tau = 0.1$.

from the main signal $x(t-\tau)$, the systems will not synchronize. On the other hand if we choose it so that it is linear in $x(t-\tau)$, it will be easy to reveal the state of the system. We therefore add to the signal $x(t-\tau)$ a small perturbation, that uses for simplicity two time-delayed states, $x(t_1)$ and $x(t_2)$, where $t_1 = t - \tau_1$ and $t_2 = t - \tau_2$, for example:

$$f(t) = x(t_1) + sgn(x(t_1))A(x(t_1) - x(t_2))^2 \tag{3}$$

The larger the amplitude A is, the harder it is to synchronize. Synchronization is possible up to the critical value $A_c \sim 0.36$. When approaching this value there is a probability close to 1 that the systems' parameters diverge. The diversion probability grows with A, and it is higher for the the attacker than for the parties. We therefore wish to enhance the security even further by using a *dynamic* amplitude A in Eq. 3, which changes dynamically throughout the synchronization process in the following way:

$$A = \frac{1}{B|f(x_1(t)) - f(x_2(t))|^\rho + C} \tag{4}$$

At first A is relatively low, so that the parties will start coming closer. Gradually they get closer and A grows so that synchronization grows harder. Because the attacker's probability to diverge is higher, using a dynamic pre-factor A affects him much more than it does the

parties [13], even if he limits himself so as not diverge, his success probability is greatly reduced.

Using a network of N>1 is essential for the security as will later be shown. What kind of network architecture should be used? In our simulations we checked several network topologies: fully connected (or "globally coupled"), networks of nearest-neighbor connections, random networks, and small-world networks. Many of these architectures were fairly secure, and the cyclic network, demonstrated in Fig. 1, was most secure of all.

Two cyclic networks 1 and 2 are coupled: each node is coupled to a parallel node in the other network and to its preceding neighbor by its x value as described in Eq. 3. The two networks exchange N signals, x^i $i = 1...N$, every time step, and use the following dynamics:

$$\frac{dx_1^i}{dt} = 10(y_1 - x_1) + K[f(x_2^i) - f(x_1^i)] + W[g(x_1^{i+1}) - g(x_1^i)] \tag{5}$$

For simplicity we use $g(x) = f(x)$ but with a static amplitude $A = 0.1$. The systems reach a state of synchronization in which although the values of the two networks are identical, there are no clusters among the nodes of each network and they are de-synchronized (if the inner coupling strength W is not too strong). The attractor dimension is N, and x^i can be utilized as an encryption-key.

Two attack strategies are considered. The first, an attacker who uses the same dynamics as the two systems, and follows their steps throughout the process, and using the same signals as they do tries to synchronize too. We name this attack the "regular following attack" (RFA). The RFA tries to follow the parties, and might use a larger K value, to increase his tracing steps [6]. Indeed when doing so his probability to synchronize increases, however he is unsuccessful when using a network, see Fig. 3.

The second attack strategy tries to reveal the state of the system by an analysis of the transmitted signals. We name this attacker the "embedded signal attack"(ESA)[16]. The ESA attack tries to analyse the transmitted signal by embedding the signal to a space defined by signals transmitted in different time steps. For example a three dimensional F-space $F\{f(t_1), f(t_2), f(t_3)\}$. The ESA tries to map an area in F-space to a range of x-values. Does a point in F-space uniquely define the current state of the system?

We claim that both attackers, RFA and ESA, are useless when using two ingredients: 1)Using a dynamic amplitude A, and 2)Using a network (N>1).

The RFA attacker tries to synchronize with the parties. When is he considered successful?

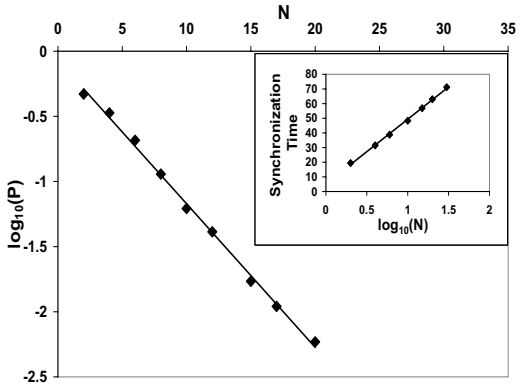

Figure 3: Semi-log plot of the probability of the RFA attacker to synchronize one node, versus N. Inset: The parties' synchronization time versus log(N). For both graphs the parties use K=8 and W=2 and the attacker uses K=14 and W=2 (see [15]), $\rho = 1.5$, $B = 200$ and C is randomly chosen in the range $[3, 4]$.

If he synchronizes all the nodes completely? Probably synchronizing only part of them is enough. We set a very soft criterion and considered a successful attacker one that manages to synchronize at least one node by only 4 digits, while the parties synchronize *all* the nodes by 7 digits. Albeit the soft criterion, we observed that the probability for an attacker to succeed decays exponentially with N, as demostrated in Fig. 3. The parties' synchronization time, on the other hand, grows only like $log(N)$, as displayed in the inset of Fig. 3.

The reason that using a network increases the security against RFA, is that every Lorenz system in the attacker's network has a probability to diverge, or in other words to have at least one positive Lyapunov exponent[17]. When using a network, if there is a node that diverges, he affects his neighbors as well, and they too start to diverge. Even if the attacker artificially limits himself not to exceed a certain value, he nevertheless loses track of the parties. It is sometimes enough even for just a few straying nodes to drag the whole network away from the right path. The attacker finds it difficult to prevent this from happening because if he cuts out even one diverging node, he is left with an open chain.

The second attack strategy (ESA) is also affected when using a network and using a dynamic A, as defined in Eq. 4. The ESA attacker defines a space $F\{f(t), f(t'), f(t")...\}$, using historical values of f from different time steps t, t', etc. and tries to map the F-space

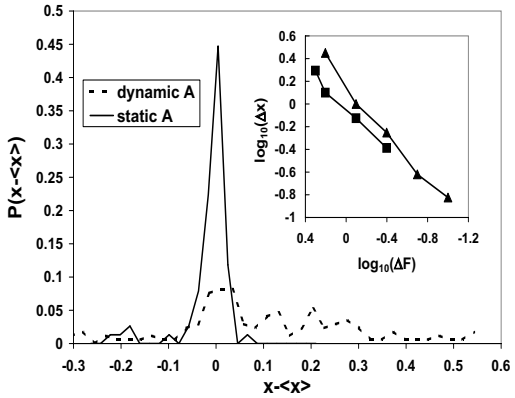

Figure 4: The ESA attack on one node, N=1. The graph shows the distribution of x values for a window in F-space, $F\{f(t), f(t-0.3), f(t-0.9)\}$, with window edge of 0.02. For dynamic A as defined in Eq.4 (dashed line) and static $A = 0.01$ (black line). Inset: A log-log plot of the width of 70% of the distribution around the average x value, Δx, vs. the window edge size in F-space, ΔF. Two cases are displayed: a 3d F-space $F\{f_1(t), f_1(t-0.3), f_1(t-0.9)\}$ for N=1 (triangles), and a 6d F-space $F\{f_1(t), f_1(t-0.3), f_1(t-0.9), f_2(t), f_2(t-0.3), f_2(t-0.9)\}$ for N=2 (squares). A static A is used in both cases, $A = 0.01$.

to corresponding x-values of the system. If for a small window in F-space there is a small corresponding range of x-values, then the mapping is possible. Yet if the distribution of x-values corresponding to a small window in F-space is wide, then x is not uniquely defined and mapping is not possible. Fig. 4 shows the case of N=1, for a dynamic and static A. When using a static A, the distribution of x-values is peaked, therefore the ESA is successful. However when using a dynamic A, the distribution of x-values corresponding to a small window in F-space is wide. Because A is dynamic there exist many close trajectories of f that lead to different x-values. However, even if the distribution of the x-value is wide, this method still provides the attacker with much information about parties' current state, for $N = 1$.

The second ingredient is therefore necessary: using a network. When using a network of size N>1, for the ESA strategy to succeed it requires an F-space dimension proportional to N. The inset in Fig. 4 displays a log-log plot of the width of the distribution of x-values vs. the size of the window edge in F-space. For a 3d F-space for N=1 and a 6d F-space for N=2,

the width of the x-distribution decreases linearly with the F-space window size. When using an F-space with $d < 6$ for N=2, the width of x-distribution reaches a constant value, when decreasing the window size. Therefore for N=2 a 6d F-space is required for mapping from F to x, and it seems that the F-space dimension should be 3N in order to reveal x. However this is true for the case of a static amplitude A, we therefore demand both a dynamic A and $N > 1$.

Other successful attack strategies that have been used to break the cryptographic systems based on synchronization of neural networks [10] are also irrelevant to this system. The most successful attack strategy, is the 'Majority Attack', which is based on an ensamble of cooperating attackers[14]. For the system presented in this letter, a group of attackers does not have a better success rate than a single attacker. Because of the *linear instability* of the chaotic dynamics, even if an attacker starts very close to the parties, because his Lyapunov exponent is positive he will not synchronize. Therefore, when using networks by the size of N, that synchronize α digits of each Lorenz system, deciphering the key requires $O(10^{\alpha N})$ attackers.

To conclude, the ability of two chaotic systems to synchronize when coupled by a time-delayed signal is used to create a cryptographic system. The signals do not reveal the state of the system, yet still enable synchronization. When expanding the system to have N degrees of freedom by weakly coupling N lorenz systems, a secure cryptographic system is constructed. Several factors contribute to the security of this system: The linear instability of the dynamics, the fact that the two parties are mutually coupled while the attacker is one-way coupled, and the structure of the network which lets individual defects affect the entire system.

Fruitful discussions with Arkady Pikovsky are acknowledged. The research if I.K is partially supported by the Israel Academy of Science.

[1] L.M.Pecora and T.L.Carroll, Phys. Rev. Lett. 64, 821 (1990).

[2] T.L.Carroll and L.M.Pecora, IEE Trans. Circuits Syst. 38, 453 (1991).

[3] L.M.Pecora and T.L.Carroll, Phys. Rev. A 44, 2374 (1991).

[4] K.Cuomo and A.Oppenheim , Phys. Rev. Lett. 71, (1993)

[5] G.Grassi and S.Mascolo, IEEE Trans. Circuits Syst. 46, 49 (1999).

[6] Synchronization: A Universal Concept in Nonlinear Scinences. Arkady Pikovsky, Michael

Rosenblum and Jurgen Kurths Cambridge U. Press, N.Y. (2001).

[7] M. Rosen-Zvi, E. Klein, I. Kanter and W. Kinzel, Phys. Rev. E **66** 066135 (2002).

[8] X.Wang, M.Zhan, C.H.Lai and H.Gang, Chaos (Dec.2003)

[9] G.Perez and H.A.Cerdeira, Phys. Rev. Lett. 74, 1970 (1995)

[10] I. Kanter, W. Kinzel and E. Kanter, Europhys. Lett., **57**, 141 (2002).

[11] F.M. Atay, J. Jost and A. Wende, Phys. Rev. Lett. 92, 144101 (2004)

[12] M. Dhamala, V.K. Jirsa and M. Ding, Phys. Rev. Lett. 92, 074104 (2004)

[13] The attacker can artificially limit his x-value so that it does not diverge. Indeed this improves his probability to synchronize, but not enough to break the security. Our results in Fig. 3 are for an attacker who limits his x-value artificially such that $|x| < 22$.

[14] L. Shacham, E.Klein, R. Mislovaty, I.Kanter and W.Kinzel, Phys. Rev. E 69, 066137 (2004).

[15] When the parties use $K = 8$, the optimal measured coupling strength for the attacker is $K \sim 14$.

[16] We thank Arkady Pikovsky for suggesting this analysis method.

[17] E. Klein, R. Mislovaty, I. Kanter and W. Kinzel (unpublished).