

## شناسایی پارامترهای مجهول سیستم آشوبی به منظور سنکرون سازی توسط الگوریتم ژنتیک

مهناز اروانه

حبیب رجبی مشهدی

مهنوش شجیعی

دانشگاه فردوسی، دانشکده مهندسی، گروه برق

E-mail: ma\_sh50@um.ac.ir

**چکیده-** این مقاله روش جدیدی را برای سنکرون سازی اسیلاتورهای آشوبی به کمک الگوریتم ژنتیک پیشنهاد می‌کند. برای پیاده سازی این روش از بین سیستم‌های آشوبگونه مختلف، سیستم راسلر انتخاب شده است. هدف از سنکرون سازی یکسان شدن پارامترهای مجهول سیستم پاسخ با سیستم راه انداز و صفر شدن خطا می باشد. در حل این مساله نیاز به یک پاسخ بهینه برای سیستم نداریم بلکه سیستم دینامیکی می‌باشد و به سیگنال کنترلی احتیاج هست. الگوریتم ژنتیک همیشه در محیطهای استاتیک استفاده می شده اما در روش ابتکاری الگوریتم ژنتیک پی در پی اجرا شده است و پارامترهای مجهول سیستم پاسخ و سیگنال کنترلی به صورت تکه ای در زیر بازه های کوچک زمانی محاسبه و با کنار هم قرار دادن آنها، مقادیر نهایی درکل زمان به دست می آید. بدین ترتیب پارامترهای متغیر با زمان برای سنکرون سازی فراهم می شوند. الگوریتم ژنتیک با قابلیت جستجوی سراسری، سعی در کاهش زمان سنکرون سازی دارد. در روش پیشنهادی در کمتر از 40 ثانیه سنکرون سازی انجام می گیرد و خطا صفر می گردد. همچنین در این مقاله اثر تغییر شرایط اولیه سیستم راه انداز نیز بررسی شده است.

**واژه های کلیدی :** سنکرون سازی، الگوریتم ژنتیک (GA)، سیستم آشوبی

nonconvex و پیچیده می باشد که می توان با استفاده از الگوریتم ژنتیک به حل آن پرداخت.

خصوصیت مشترک سیستم‌های آشوبگونه داشتن رفتار غیر قابل پیش‌بینی و حساسیت بسیار زیاد به شرایط اولیه است بطوریکه با کوچکترین تغییر در شرایط اولیه پاسخها بسیار متفاوت خواهند شد. رفتارهای آشوبگونه را می‌توان در بسیاری از سیستم‌های فیزیکی مشاهده کرد [2].

جستجو در فضاهای پیچیده با تعداد ابعاد بالا، غیر خطی بودن، بهینه سازی چند هدفی و قابلیت داشتن محاسبات موازی از مزایای الگوریتم ژنتیک می باشد.

همزمان سازی سیستم‌های آشوبگونه کاربردهای زیادی در زمینه‌های مختلف علوم مانند کاربرد در لیزر، راکتورهای شیمیایی، اقتصاد کلان، مخابرات ایمن و بیولوژی دارد [2]. دینامیکهایی نظیر سیستم تنفسی، فعالیت قلب، سیستم‌های اکولوژیکی و ... خواص سنکرون از خود نشان

### 1- مقدمه

از زمانی که Pecora و Carroll روشی را برای همزمان سازی دو سیستم آشوبگونه با شرایط اولیه مختلف معرفی کردند [1]، همزمان سازی آشوب به عنوان یک مسأله مهم در مبحث سیستم‌های غیر خطی به طور گسترده‌ای وسعت یافت. همزمان سازی از مهمترین زیر شاخه‌های کنترل آشوب می‌باشد. در سال‌های اخیر سنکرون کردن سیستم‌های آشوبگونه مورد توجه زیادی قرار گرفته است و روش‌های گوناگونی هم برای سنکرون سازی آشوب پیشنهاد و ارائه شده است [8]. که می توان به روشهای کنترل تطبیقی، کنترل مد لغزشی، کنترل بازگشت به عقب اشاره نمود. اما مساله سنکرون سازی توسط الگوریتم ژنتیک تا به حال مورد توجه قرار نگرفته است.

مساله سنکرون سازی سیستم آشوبی مساله ای غیرخطی،

به جذب کننده آشوبی این سیستم می باشد. دو سیستم آشوب گونه به فرم راسلر [3] را که با معادلات زیر بیان می شوند به عنوان master (drive) و slave (response) در نظر می گیریم.

$$\text{drive: } \begin{cases} \dot{x}_1 = -x_2 - x_3, \\ \dot{x}_2 = x_1 + a_1 x_2, \\ \dot{x}_3 = a_2 + x_3(x_1 - a_3), \end{cases} \quad (1)$$

$$\text{response: } \begin{cases} \dot{y}_1 = -y_2 - y_3 + u_1, \\ \dot{y}_2 = y_1 + \hat{a}_1(t)y_2 + u_2, \\ \dot{y}_3 = \hat{a}_2(t) + y_3(y_1 - \hat{a}_3(t)) + u_3, \end{cases} \quad (2)$$

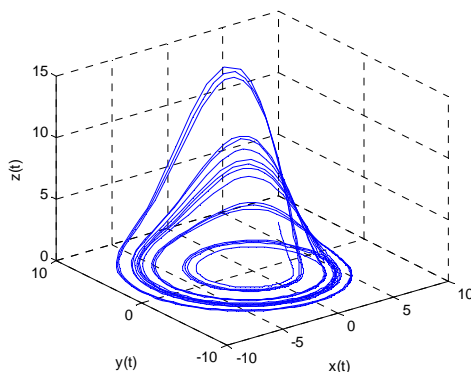
این دو سیستم سه بعدی و دارای حالت های  $y(t), x(t) \in R^3$  می باشند.

خطای سیستم به صورت تفاضل حالت های دو سیستم راه انداز و پاسخ تعریف میشود [7]:

$$e_i = y_i - x_i \quad i = 1, 2, 3 \quad (3)$$

بنابراین دینامیک خطای سیستم به صورت زیر در می آید:

$$\begin{aligned} \dot{e}_1 &= -e_2 - e_3 + u_1 \\ \dot{e}_2 &= e_1 + \hat{a}_1 e_2 + (\hat{a}_1 - a_1)x_2 + u_2 \\ \dot{e}_3 &= \hat{a}_2 - a_2 + e_1 e_3 + x_3 e_1 + x_1 e_3 - \hat{a}_3 e_3 + (\hat{a}_3 - a_3)x_3 + u_3 \end{aligned} \quad (4)$$



شکل 1- جذب کننده آشوبی راسلر برای سیستم با پارامترهای

$$a_1 = a_2 = .2, a_3 = 4.7 \text{ و شرایط اولیه}$$

$$(x_1(0), x_2(0), x_3(0)) = (.3, 2.5, 3.2)$$

می دهند. به طور کلی مسأله همزمان سازی سیستم های آشوب گونه به این معنی است که مسیرهای حالت دو سیستم آشوب گونه به طور یکسان و همزمان با یکدیگر نوسان کنند. شایان ذکر است که پدیده آشوب تنها در سیستم های غیر خطی وجود دارد.

در مسأله سنکرون سازی، یک سیستم آشفته بعنوان راه انداز و سیستم آشفته دیگر بعنوان پاسخ در نظر گرفته می شود [4] و فرض بر این است که دو سیستم کاملاً یکسان می باشند. آنگاه یک سیگنال خروجی از سیستم راه انداز جهت راه اندازی سیستم پاسخ ارسال می گردد. حال باید با استفاده از این سیگنال و اعمال یک کنترل مناسب بر روی آن، متغیرهای حالت سیستم پاسخ پس از یک زمان گذر، به متغیرهای حالت سیستم راه انداز همگرا شوند.

به کار بردن روش الگوریتم ژنتیک باعث حذف پروسه سعی و خطا برای تخمین پارامترهای مجهول می شود.

حتی می توان اثر نویز روی سیستم را نیز در نظر گرفت و از الگوریتم ژنتیک استفاده نمود. زیرا GA در محیط های نویزی نیز به خوبی عمل می نماید.

از بین سیستم های آشوبی زیادی که وجود دارد، سیستم راسلر انتخاب شده است.

در این مقاله با استفاده از الگوریتم ژنتیک به سنکرون سازی سیستم راسلر و یکسان شدن پارامترهای مجهول سیستم پاسخ با سیستم راه انداز و صفر شدن خطا پرداخته شده است.

در ادامه، بخش دوم به معرفی سیستم راسلر و بخش سوم به معرفی کوتاهی از الگوریتم ژنتیک و ارائه روش پیشنهادی در حل مسأله سنکرون سازی اختصاص دارد. در بخش چهارم نتایج شبیه سازی روش پیشنهادی بیان می گردد و در نهایت، بخش آخر به نتیجه گیری و جمع بندی پرداخته می شود.

## 2- سیستم راسلر و جذب کننده های آشوبی

سیستم راسلر مدل ساده شده ای از دینامیک واکنش های شیمیایی در یک تانک همزن ارائه می کند شکل 1 مربوط

پارامترهای سیستم راه انداز  $(a_1, a_2, a_3)$  معلوم می باشد ولی پارامترهای متغیر با زمان سیستم پاسخ  $(\hat{a}_1(t), \hat{a}_2(t), \hat{a}_3(t))$  مجهول می باشند.

هدف از سنکرون سازی یکسان شدن پارامترهای مجهول سیستم پاسخ با سیستم راه انداز و صفر شدن خطا می باشد.

$$\hat{a}_1 = a_1, \quad \hat{a}_2 = a_2, \quad \hat{a}_3 = a_3 \quad (5)$$

$$e_1 = e_2 = e_3 = 0$$

در این مقاله برای دستیابی به این هدف، ورودیهای کنترلی  $u_i$  به گونه ای انتخاب می شوند که شرط زیر بر آورده گردد:

$$\lim_{t \rightarrow \infty} \|e(t)\| = 0 \quad (6)$$

### 3- الگوریتم ژنتیک (GA)

الگوریتم ژنتیک یک روش بهینه سازی بر اساس جستجو است که توسط جان هولند [5] در سال 1962 ارایه شد و از دو اصل انتخاب و تولید نسل در طبیعت بهره برده است [6]. از الگوریتم ژنتیک در کاربردهای مختلفی مانند بهینه سازی توابع، شناسایی سیستمها، پردازش تصویر و ... استفاده می شود. با اینکه دامنه کاربرد این الگوریتم بسیار وسیع است اما اغلب به آن به چشم بهینه سازی توابع نگاه می شود [6].

هنگامی که هدف پیدا کردن جوابی نزدیک به پاسخ بهینه برای یک تابع چند بعدی با چندین قید باشد بهینه سازی ژنتیکی بسیار موثر و کارا هستند. در این روش بر اساس قاعده بقای ژنها یا کروموزومهایی که بیشترین تطابق با محیط را دارند و همچنین تبادل تصادفی اطلاعات ژنتیکی، مجموعه جدیدی از کروموزومها تولید می شوند که تشکیل دهنده نسل بعدی هستند. بدین ترتیب با پیشرفت تدریجی الگوریتم، نسلهای بهتر و سازگارتری پدید می آیند. به عبارت دیگر نقاط بهتری از فضای جستجو پیدا می شوند.

GA جستجو را بر اساس جمعیتی از نقاط انجام میدهد نه یک نقطه. تعداد این نقاط نیز مهم است. به این صورت که تعداد جمعیت زیاد، محاسبات را طولانی کرده و کندتر

به جواب بهینه نزدیک میشود و تعداد جمعیت کم فضای جستجو را به خوبی پوشش نمی دهد و ممکن است الگوریتم در یک جواب محلی گرفتار شود. در این الگوریتم نیاز به ارزیابی افراد جمعیت می باشد و این ارزیابی با یک تابع، موسوم به تابع معیار انجام می گیرد. افرادی که نمره بالاتری کسب نموده اند شانس بقا و تولید مثل بیشتری پیدا می کنند.

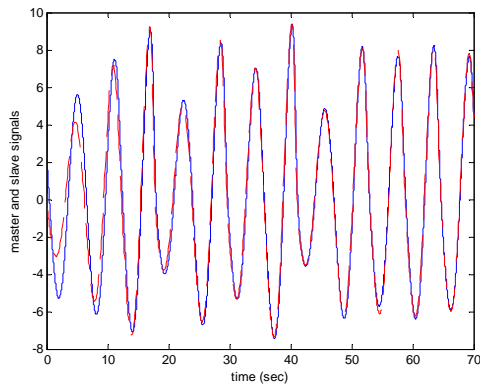
از مزایای دیگر الگوریتم ژنتیک این است که حساسیتی نسبت به محدودیتهای خاص نظیر پیوستگی و مشتق پذیری تابع هزینه ندارد.

در این تحقیق به جای کد کردن باینری از روش کد کردن حقیقی استفاده شده است. زمان اجرای برنامه در کد کردن با اعداد حقیقی از کد کردن باینری کوتاهتر می باشد. سایز جمعیت اولیه 100 و تعداد نسلها 1000 و احتمال برش 95 درصد انتخاب شده است.

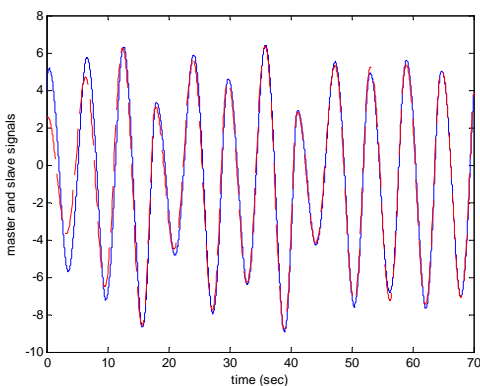
در این مساله پارامترهای  $\hat{a}_i$  و  $u_i$  متغیر با زمان می باشند. برای اینکه بتوان این مساله را با الگوریتم ژنتیک حل نمود زمان به زیر بازه های 0.1 ثانیه تقسیم شد و  $\hat{a}_i$  و  $u_i$  ها در هر زیر بازه دارای مقداری ثابت می باشند. بدین ترتیب با به دست آمدن  $\hat{a}_i$  و  $u_i$  در هر زیر بازه و کنار هم قرار دادن آنها،  $\hat{a}_i$  و  $u_i$  نهایی درکل زمان به دست می آید. این الگوریتم برای 700 زیر بازه که معادل 70 ثانیه است اجرا شده است.

در روش پیشنهادی این مقاله مقادیر نهایی هر زیر بازه به عنوان شرط اولیه و مقادیر  $\hat{a}_{i-1}$  و  $u_{i-1}$  به عنوان جمعیت اولیه برای زیر بازه بعدی انتخاب گردید. این امر سبب میشود تا GA نقطه جستجو بعدی را از نقطه بهتری آغاز نماید و سریعتر به جواب همگرا شود. یعنی با این گونه انتخاب، پس از مدت زمانی،  $\hat{a}_i$  سریعتر به  $a_i$  میل کرده و  $u_i$  ها صفر می شوند. در این زمان سنکرون سازی به طور کامل انجام گرفته است و خطا صفر می گردد.

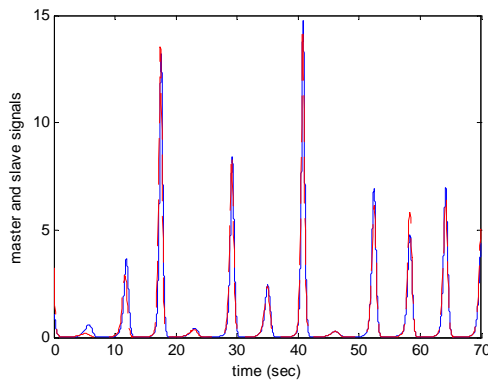
به منظور صفر کردن خطا تابع معیار انتخاب شده در هر زیر بازه به صورت زیر می باشد.



شکل 2- سیگنالهای راه انداز و پاسخ برای  $x_1, y_1$



شکل 3- سیگنالهای راه انداز و پاسخ برای  $x_2, y_2$



شکل 4- سیگنالهای راه انداز و پاسخ برای  $x_3, y_3$

در شکل 5 تمام حالت‌های دو سیستم را نشان می‌دهد که با یکدیگر سنکرون شده‌اند. این شکلها به وضوح سنکرون شدن دو سیستم راه انداز و پاسخ راسلر را در زمانی کمتر از 40 ثانیه نشان می‌دهد. و نشان دهنده این می‌باشد که سنکرون سازی پایدار می‌باشد.

$$fitness(j) = \int_{\frac{j-1}{10}}^{\frac{j}{10}} \sum_{i=1}^3 e_i^2 / 3 dt \quad (7)$$

$fitness(j)$  نشان دهنده تابع معیار در زیر بازه  $j$ ام می‌باشد.

همان گونه که در شکلها مشاهده می‌شود با استفاده از روش ارایه شده سنکرون سازی به طور کامل و به خوبی انجام گرفته است.

#### 4- شبیه سازی ها

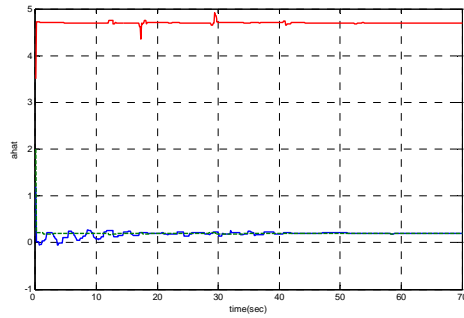
در این بخش به منظور نشان دادن قابلیت‌های روش کنترل پیشنهادی، شبیه سازی سنکرون کردن دو سیستم نمونه، مد نظر قرار گرفته است.

در این مثال از دو سیستم آشوب گونه راسلر (Rossler) استفاده شده است. معادلات توصیفی این سیستم به صورت زیر می‌باشد.

$$\begin{cases} \dot{x}_1 = -x_2 - x_3, \\ \dot{x}_2 = x_1 + 0.2x_2, \\ \dot{x}_3 = 0.2 + x_3(x_1 - 4.7), \end{cases} \quad (8)$$

شرایط اولیه را برای سیستم master به صورت  $(x_1(0), x_2(0), x_3(0)) = (0.3, 2.5, 3.2)$  و برای سیستم slave به صورت  $(y_1(0), y_2(0), y_3(0)) = (2.5, 4, 1)$  و برای پارامترهای مجهول متغیر با زمان سیستم پاسخ، به صورت  $(\hat{a}_1(0), \hat{a}_2(0), \hat{a}_3(0)) = (1.2, 2, 3.5)$  انتخاب می‌نمائیم.

در شکل 2 تا 4 هر یک از حالت‌های  $x_i, y_i$  نشان داده شده است که توسط روش پیشنهادی سنکرون شده‌اند.



شکل 7 - پارامترهای مجهول سیستم پاسخ

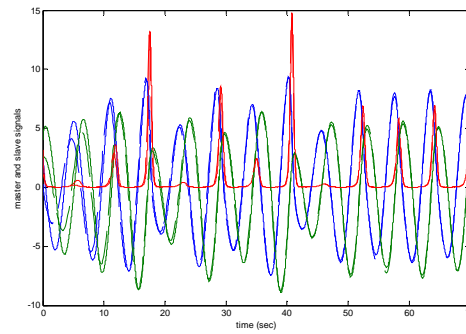
شکل 7 پارامترهای مجهول سیستم پاسخ یعنی  $\hat{a}_i$  را نشان می دهد که دقیقاً به مقادیر  $a_1 = a_2 = .2, a_3 = 4.7$  سیستم راه انداز همگرا شده اند که نشان دهنده سنکرون شدن کامل سیستم پاسخ با سیستم راه انداز می باشد.

شرایط اولیه از مقادیر اولیه  $(0.3, 2.5, 3.2) = (x_1(0), x_2(0), x_3(0))$  به مقادیر  $(0.4, 2.6, 3.3)$  افزایش داده شد و زمان سنکرون سازی از 40 ثانیه به 80 ثانیه تغییر یافت و همچنین شرایط اولیه از مقادیر اولیه به مقادیر  $(0.2, 2.4, 3.1)$  کاهش داده شد و زمان سنکرون سازی از 40 ثانیه به 70 ثانیه افزایش یافت که می توان نتیجه گرفت که زمان همگرا شدن الگوریتم ژنتیک تحت تاثیر شرایط اولیه می باشد پس باید در انتخاب شرایط اولیه نیز دقت شود.

همچنین با این روش سعی در سنکرون سازی سیستم پاسخ با سیستم درایو تنها با یک ورودی کنترلی به جای سه ورودی کنترلی شد. نتایج نشان داد این الگوریتم ژنتیک مانند روشهای کلاسیک قادر به سنکرون سازی تنها با یک سیگنال کنترلی نمی باشد.

## 5- نتیجه گیری

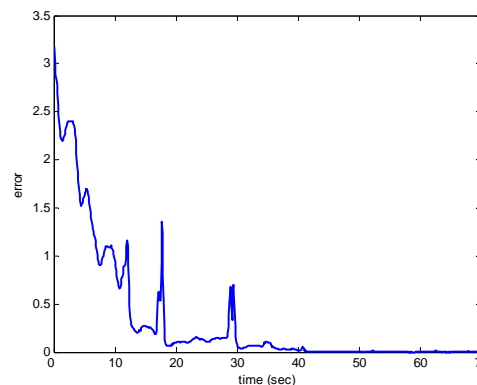
در این مقاله روش جدیدی برای سنکرون سازی اسلاتورهای آشوبی به کمک الگوریتم ژنتیک پیشنهاد شده است. سیستم راسلر با پارامترهای غیر قطعی بررسی شد. پارامترهای غیر قطعی سیستم راسلر با روش پیشنهادی به دست آمدند. در این تحقیق به منظور سنکرون سازی سیستم های آشوبی به روش الگوریتم ژنتیک دو نوآوری ارائه گردیده است:



شکل 5- سیگنالهای راه انداز و پاسخ

شکل 6 نشان دهنده خطا است که بعد از مدتی کاملاً صفر شده است.

جهشهای مشاهده شده در شکل 6 ناشی از خطای بین  $x_3, y_3$  می باشد. تغییرات زیاد و سریع  $x_3$  باعث می شود یافتن نقاط نوک تیز برای الگوریتم ژنتیک دشوار و همراه با خطا باشد.



شکل 6- سیگنال خطا

با توجه به اینکه خطا بعد از 40 ثانیه که معادل 400 زیر بازه است کاملاً صفر گردیده، می توان نتیجه گرفت  $\hat{a}_i$  به  $a_i$  میل کرده و دیگر بدون نیاز به  $u_i$  کنترلی، دو سیستم به صورت سنکرون با یکدیگر نوسان می کنند.

به دلیل متغیر با زمان بودن پارامترهای  $\hat{a}_i$  و  $u_i$  ،  $i = 1, 2, 3$  ، زمان به زیر بازه های 0.1 ثانیه تقسیم شد و  $u_i$  و  $\hat{a}_i$  ها در هر زیر بازه دارای مقداری ثابت می باشند.

مقادیر نهایی هر زیر بازه به عنوان شرط اولیه و مقادیر  $u_{i-1}$  و  $\hat{a}_{i-1}$  به عنوان جمعیت اولیه برای زیر بازه بعدی انتخاب گردید که این امر سبب میشود تا GA نقطه جستجو بعدی را از نقطه بهتری آغاز نماید و سریعتر به جواب همگرا شود. در شکلها مشاهده می شود که با استفاده از روش ارایه شده سنکرون سازی به طور کامل و به خوبی در کمتر از 40 ثانیه انجام گرفته است.

به دلیل استفاده از الگوریتم ژنتیک با قابلیت جستجو موازی در فضای با ابعاد متعدد، امکان توسعه آن برای سنکرون سازی همزمان چندین سیستم آشوبی با یکدیگر وجود دارد.

استفاده از الگوریتم ژنتیک این امکان را می دهد که برای مقادیر پارامترهای  $\hat{a}_i$  و  $u_i$  ها محدودیتهایی در نظر بگیریم. با استفاده از این شرط می توان از ایجاد سیگنال کنترلی با دامنه بزرگ جلوگیری نمود.

## مراجع

- [1] Pecora L. and Caroll T., "Synchronization in Chaotic Systems", Phys. Rev. Lett., Vol. 64, No.8, 1990, PP. 821-824.
- [2] Boccaletti S., Grebogi C., Lai Y. C., Mancini H., Maza D., "The control of chaos: theory and applications", Physics Reports, Vol.329, 2000, PP. 103-197.
- [3] Rössler O.E., "An equation for continuous chaos", Phys. Lett. A., Vol. 57, 1976, PP. 397-398.
- [4] Wang S.X., "simulation of chaos synchronization" , London, 1998
- [5] Holland J., "adaptation in natural and artificial systems", University of Michigan Press, 1975
- [6] Whitley D., "A Genetic algorithm tutorial", Statistics and computer, Vol. 4, pp.65-85, 1994
- [7] Boccaletti S, Osipov G, Valladares D.L, and Zhou C.S "The synchronization of chaotic systems" , Phys.Reports, vol.1, No.366, pp.1-101, 2002
- [8] Lü J., Chen G., Zhang S., "Dynamical analysis of a new chaotic attractor", Int. J. Bifurcation. Chaos, Vol. 12, No. 5., PP. 1001- 1015, 2002.