International Journal of Electrical, Electronic and Communication Sciences
ISSN: 2517-9438
Vol:3, No:11, 2009

Particle filter applied to noisy synchronization in polynomial chaotic maps

Moussa Yahia, Pascal Acco and Malek Benslama

Abstract—Polynomial maps offer analytical properties used to obtain better performances in the scope of chaos synchronization under noisy channels. This paper presents a new method to simplify equations of the Exact Polynomial Kalman Filter (ExPKF) given in [1]. This faster algorithm is compared to other estimators showing that performances of all considered observers vanish rapidly with the channel noise making application of chaos synchronization intractable. Simulation of ExPKF shows that saturation drawn on the emitter to keep it stable impacts badly performances for low channel noise. Then we propose a particle filter that outperforms all other Kalman structured observers in the case of noisy channels.

Keywords—Chaos synchronization, Saturation, Fast ExPKF, Particle filter, Polynomial maps.

I. INTRODUCTION

CHAOTIC synchronization under noisy channel played a key role during last decade in chaotic telecommunication systems. First theoretical works on chaos synchronization neglected noise considerations [2], [3]. Then, the idea to use coupled chaotic oscillators in telecommunication is introduced [4], [5]. Additive noise in the channel destroys synchronization properties and rises the problem of noise cleaning. As performances quickly decay in presence of noise other communication schemes, as non-coherent and impulse synchronization, where considered to avoid synchronization. Kolumbán et. al give a review of communication schemes and performance limits in three papers [6], [7], [8].

The problem of synchronization takes roots in control system theory and can be seen as the state estimation of a stochastic non-linear system.

A. Kalman structured observers

Kalman filtering can be applied to synchronize systems. In the discrete time and linear case, the emitter state x_k is modeled by a linear dynamical function $f(x)$, with additive dynamic noise $\eta_k \sim \mathcal{G}(0,Q)$ [1] and measurement noise $\nu_k \sim \mathcal{G}(0,R)$:

$$\begin{cases} x_{k+1} &= f(x_k) + \eta_k \\ y_k &= h(x_k) + \nu_k \end{cases} \tag{1}$$

The linear measurement function $h(x)$ and measurement noise ν_k represent the channel model and channel noise respectively. As the dynamic noise η_k represents real noise in the emitter, or approximate numerical noise encountered in the simulation or calculator, the initial state x_0 is a random variable with pdf[2] $p(x_0) = \mathcal{G}\left(\bar{x_0}, \sigma_x^2\right)$.

The Kalman filter constructs an estimated state $\hat{x_k}$ knowing the measurements y_k at the receiver. This receiver is optimal in regard of the mean square error criteria (MSE)

$$MSE = \frac{1}{N}\sum_{k=0}^{N}(x_k - \hat{x_k})^2 \tag{2}$$

The structure of the Kalman observer needs to estimate mean, variance and covariance of stochastic variables:

$$\begin{cases} \hat{x}_{k+1} &= \tilde{x}_{k+1/k} + K_{k+1}\left(y_{k+1} - \tilde{y}_{k+1/k})\right]) \\ P_{k+1} &= P_{k+1/k} - K_{k+1}^2\, P_{y_{k+1/k}\, y_{k+1/k}} \\ K_{k+1} &= \frac{P_{x_{k+1/k}\, y_{k+1/k}}}{P_{y_{k+1/k}\, y_{k+1/k}}} \end{cases} \tag{3}$$

For linear functions f and h, expressions 3 are analytical and easy to compute, whereas many techniques where developed to deal with non-linear functions f and h.

The *Extended Kalman filter* (EKF) uses successive linear approximations of the mean and variance statistics to build its state estimates [9][10]. Strongly non-linear systems, like chaotic dynamics, approximations generate large estimation errors which limit the use of EKF to weakly nonlinear systems.

The *Unscented Kalman Filter* (UKF) was introduced [11], [12] to solve this problem. The Gaussian random state is represented using a minimal number of samples, called sigma points. At each iteration, the UKF propagate the sigma points through the true nonlinear function and computes the posterior mean and covariance approximation. The approximation of higher order moments is enabled by changing the sigma points weights. This solution is advantageous to cope with strongly nonlinear system and to avoid computation of function derivative as for the EKF.

Norgaard et al. [13] use polynomial interpolation of the dynamical function f and exploit Stirling's formula to obtain the mean and covariance of the state distribution. This nonlinear transformation is then exploited in a recursive manner owing to a Kalman Filter structure.

More recently, Luca et al [1] continue this work and propose a closed-form state estimator named *Exact Polynomial Kalman Filter*(ExPKF) for polynomial nonlinear system with f and h polynomial of respective order N and M.

ExPKF can then synchronize to a chaotic polynomial map with optimal performances. Still is the problem of emitter stability when noise is added to the chaotic recurrence.

M. Yahia is with the Department of Electronic, Faculty of Science and Technology, Jijel University, 18000, Algeria e-mail: yahiamos@yahoo.fr.

P. Acco is with LATTIS, University of Toulouse, France,

M. Benslama is with Univesity of Maitouri Constantine

Manuscript received April 19, 2009; revised June 16, 2009.

[1] $x \sim \mathcal{G}(m,v)$ means that the random variable x has a Gaussian probability density function of mean m and variance v

[2] probability density function

B. Emitter stability

For secure and/or wide-band communication purpose, the emitter iterative map $T : x_{k+1} = f(x_k)$ is placed in an attractive basin of a chaotic strange attractor. We will focus on chaotic system whose attractive basin is a simple connex set $U = [a, b]$. Then chaotic properties guarantee that the map is onto, i.e. $f(U) \subset U$, and there is no divergent trajectories.

Stability problems arise when the dynamic noise η_k is added to $f(x_k)$. The noise can then push the stable state x_k out the attractive basin and start to follow a divergent trajectory.

An easy solution is to saturate to the set U each iteration of the map. In this case the dynamic noise η_k in 1 is replaced with:

$$\eta_k^\star = \begin{cases} \eta_k & \text{when } \eta_k \in [a - f(x_k), b - f(x_k)] \\ a & \text{when } \eta_k < a - f(x_k) \\ b & \text{when } \eta_k > b - f(x_k) \end{cases} \quad (4)$$

Whatever the noise η_k is, the system still remains in the attractive basin U. A counterpart of this solution is that the dynamic noise $\eta_k^\star$ pdf is no more Gaussian and no more uncorrelated to the state x_k.

C. Contents

In section II we give a simpler expression of mean, variance and covariance of the ExPKF that leads to a faster algorithm than [1]. Section III compares synchronization performances and shows the impact of saturation effects. To take into account emitter saturations a particle filter is introduced in the last section.

II. Faster ExPKF algorithm

In the paper [1], the authors use full Taylor series expansion to obtain exact expression of mean, variance and covariance of any random variable distribution. In this section we will compute directly the necessary statistics values of the transformed random variable through a polynomial function. Using our simpler expressions we obtain an ExPKF faster to compute.

A. Statistics through polynomial transformation

Consider a single dimensional polynomial form:

$$y = g(x) = \sum_{n=0}^{N} a_n x^n \quad (5)$$

In order to find the statistical values of a transformed random variable y we first focus on terms ($z = x^n$), the initial independent distribution x can written by the following form

$$x = \bar{x} + \delta x \quad (6)$$

where δx is the zero mean random variable extracted from the initial random variable x of mean $\bar{x}$. By mean of Pascal's triangle the terms x^n can then be expanded to:

$$x^n = \sum_{i=0}^{n} C_n^i \, \bar{x}^{n-i} \, \delta x^i \quad (7)$$

where $C_n^i = \frac{n!}{i!(n-i)!}$, is the combinatorial function, consequently the expression 5 becomes:

$$y = \sum_{n=0}^{N} a_n \sum_{i=0}^{n} C_i^n \, \bar{x}^{n-i} \delta x^i \quad (8)$$

The mean $\bar{y} = E[y] = E[g(x)]$ can be expressed as

$$\bar{y} = \sum_{n=0}^{N} a_n \sum_{i=0}^{n} C_i^n \, \bar{x}^{n-i} m_i \quad (9)$$

where m_i denote the ith-order moment of the random variable δx. In order to facilitate the computation this last Equation (9) is written in a more compact form as

$$\bar{y} = a_{0:N}^T \mathbf{C}_{0:N}^{\bar{x}} m_{0:N}^x \quad (10)$$

where $a_{0:N}$ is the polynomial coefficient vector $[a_0, ..., a_N]^T$, $m_{i:j}^x = [m_i, ..., m_j]^T$ and $\mathbf{C}_{i:j}^{\bar{x}}$ denoting a lower triangular matrix computed such as

$$\mathbf{C}_{i:j}^{\bar{x}} = \begin{bmatrix} C_i^0 \bar{x}^0 & 0 & 0 & ... & 0 \\ C_{i+1}^0 \bar{x}^1 & C_{i+1}^1 \bar{x}^0 & 0 & ... & 0 \\ C_{i+2}^0 \bar{x}^2 & C_{i+2}^1 \bar{x}^1 & C_{i+2}^2 \bar{x}^0 & ... & 0 \\ ... & ... & ... & ... & ... \\ C_j^0 \bar{x}^j & C_j^1 \bar{x}^{j-1} & C_j^2 \bar{x}^{j-2} & ... & C_j^j \bar{x}^0 \end{bmatrix} \quad (11)$$

Variance $\sigma_y^2 = E[y^2] - \bar{y}^2$ is expressed from expectation of polynomial y^2 which coefficient vector noted $e_{0:2N}$ is obtained by the convolution product of the vector $a_{0:N}$ with itself:

$$e_{0:2N} = a_{0:N} * a_{0:N} \quad (12)$$

then the term $E[y^2]$ is the same calculus than (9)applied to the polynomial vector (12)

$$E[y^2] = \sum_{n=0}^{2N} e_n \sum_{i=0}^{n} C_n^i \bar{x}^{n-i} m_i \quad (13)$$

finally we get the variance σ_y^2 in the compact matrix form

$$\sigma_y^2 = e_{0:2N}^T \mathbf{C}_{0:2N}^{\bar{x}} m_{0:N}^x - \bar{y}^2 \quad (14)$$

Covariance P_{xy} between the two variables x and y is obtained from the polynomial xy whose coefficient are $d_{0:N+1}^T = [0, a_0, ..., a_N]^T$:

$$\begin{aligned} P_{xy} &= E[(x - \bar{x})(y - \bar{y})] E[xy] - \bar{x}\bar{y} \\ &= d_{0:N+1}^T \mathbf{C}_{0:N+1}^{\bar{x}} m_{0:N+1}^x - \bar{x}\bar{y} \end{aligned} \quad (15)$$

In the case where the estimates are supposed Gaussian; $m_0 = 1$, $m_1 = 0$ and for any $k > 1$,

$$m_k = \begin{cases} k-1)\sigma_x^2 m_{k-2} & if\ k\ is\ even \\ 0 & otherwise \end{cases} \quad (16)$$

where σ_x^2 is the variance of the priori stat x, then the moments vector is given by:
$\mathrm{m}_{0:N}^x = [m_0, m_1, m_2, m_3, m_4, ...] = [1, 0, \sigma_x^2, 0, 3\sigma_x^4, ...]^T$

B. Algorithm complexity

It is necessary to know in general, how much computation time is involved in implementing the algorithm, in other words to make a complexity calculation before passing to the execution on the CPU. The purpose of our redraw of the expressions is to minimize the number of operations to generate the required statistical values of the transformed random variable via the polynomial function. In the latter work (see [1]) and for an N order polynomial map, the matrix multiplication evidently requires N^4 multiplications, plus a smaller number of operations to compute the variance σ_y^2. So this ExPKF Algorithm appears to be an $O(N^4)$ process. In our development it is easy to observe through the expressions (10), (14) and (15) that the ExPKF can, in fact, be computed in $O(N^2)$ operation with an algorithm called the $ExPKF\ Fast\ Algorithm$ which will be mentioned bellow.

C. ExPKF fast algorithm

Those previous simple expressions developed in (II-A) are then used to complete the prediction step of the ExPKF. This step computes the predicted statistics used to complete the update step (3).

The mean and the covariance of the predicted state are obtained with (10) and (14):

$$\tilde{x}_{k+1/k} = E[f(x_k) + \eta_k] = a_{0:N}^T \, \mathbf{C}_{0:N}^{\hat{x}_{k/k}} \, m_{0:N}^{x_{k/k}} \qquad (17)$$

$$\begin{aligned} P_{k+1/k} &= E[(x_{k+1/k} - \tilde{x}_{k+1/k})^2] \\ &= (e_{0:2N})^T \mathbf{C}_{0:2N}^{\hat{x}_k} m_{0:2N}^{x_k} - \tilde{x}_{k+1/k}^2 + Q \qquad (18) \end{aligned}$$

The predicted observation, the transition/innovation covariances and the variance are obtained with (10), (15) and (14):

$$\tilde{y}_{k+1/k} = E[h(x_{k+1/k})] = b_{0:M}^T \, \mathbf{C}_{0:M}^{\tilde{x}_{k+1/k}} \, m_{0:M}^{x_{k+1/k}} \qquad (19)$$

$$P_{x_{k+1/k}\, y_{k+1/k}} = d_{0:M+1}^T \, \mathbf{C}_{0:M+1}^{\tilde{x}_{k+1/k}} \, m_{0:M+1}^{x_{k+1/k}} - \tilde{x}_{k+1/k} \tilde{y}_{k+1/k} \quad (20)$$

$$P_{y_{k+1/k}\, y_{k+1/k}} = (e_{0:2M})^T \, \mathbf{C}_{0:2M}^{\tilde{x}_{k+1/k}} \, m_{0:2M}^{x_{k+1/k}} - \tilde{y}_{k+1/k}^2 + R \quad (21)$$

III. Saturation Impact on Synchronization Performance

Luca et al. compare synchronization performances of the EKF, UKF and the Scaled Uncented Kalman Filter (ScUKF) to the ExPKF. Figure 7 in [1] gives the normalized mean square error MSE/R obtained with those filters for a fourth order Chebychev polynomial $f(x) = T_4(x) = 8x^4 - 8x^2 + 1$. We note that for noise variance $R > 10^{-2}$ all filters performances tend to a limit $MSE/R \approx 0.94$. That means we can expect synchronization only for very clean channels. Luca's simulations are done with a dynamic noise $Q = \frac{R}{10}$ which do not help to distinguish whether the dynamic noise or measurement noise limits the MSE/R performances to 0.94.

Moreover, Section I-B stands that dynamic noise η_k generates saturation at the emitter. To seek the real impact of saturations, Fig. 2 shows performances with Q and R varying independently using our fast ExPKF algorithm.

Saturations occur more and more frequently as noise variance Q increase. The hypothesis of Gaussian dynamic noise

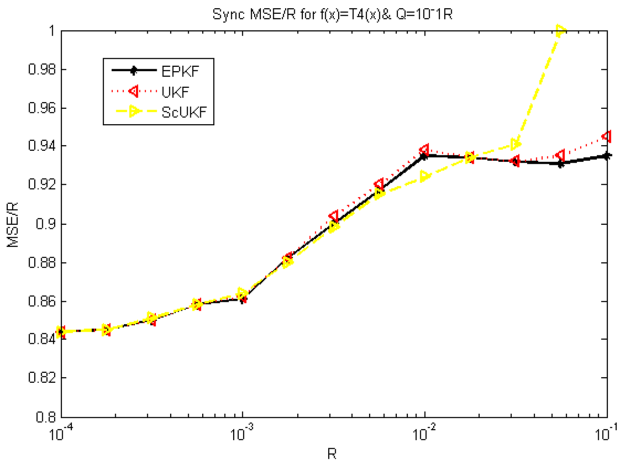

Fig. 1. Synchronization MSE/R for $f(x) = T_4(x)$. UKF and Scaled Uncented Kalman Filter (ScUKF) data are taken directly from [1]. ExPKF performances are computed with our faster ExPKF algorithm in the same conditions: $x_0 \sim \mathcal{G}(0.5, 0.26)$, $Q = \frac{R}{10}$, 10^3 transition iterations, $N = 10^5$.

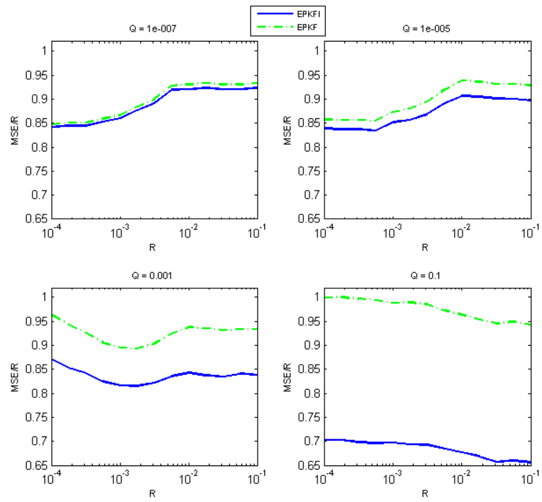

Fig. 2. MSE/R performances of our fast ExPKF algorithm and ExPKFI: $x_0 \sim \mathcal{G}(0.5, 0.26)$, $Q = \frac{R}{10}$, 10^3 transition iterations, $Nsamples = 10^5$. For small value of Q performances of the fast ExPKF algorithm gives the same results as in Fig.7 of [1] which validate the fast algorithm equations.

η_k is not met and turn this noise into a non Gaussian correlated noise $\eta_k^\star$ of equation 4. Then the ExPKF becomes under optimal. The Exact Polynomial Kalman Filter Initialized ExPKFI is then introduced as an optimal filter which can handle saturations. This filter uses an ideal second channel to transmit the information $s(k)$ whether or not a saturation at step time k exists. This filter is given as a theoretical boundary to measure performance losses due to saturation.

The ExPKFI operates like the ExPKF when there is no saturation $s(k) = 0$. When an upper/lower saturation happens $s(k) = a$ or b, then the filter is re-initialized to $\hat{x}_k = s(k)$ with estimated covariance $P_{x_k/y_k} = 0$ as the state is perfectly known at each saturation.

Fig. 2 leads to the following remarks:

International Journal of Electrical, Electronic and Communication Sciences
ISSN: 2517-9438
Vol:3, No:11, 2009

1) Whatever the dynamic noise is, the filter tends to its performance limit $MSE/R \approx 0.94$ when the measurement noise variance R is greater than 10^{-2}. So the performance limits for high dynamical noise is not linked to saturation effects.
2) For low dynamic noise $\left(0 \; to \; 10^{-7}\right)$ the ExPKFI filter is similar to ExPKF as long as very few saturation appears during those simulations. Higher dynamic noise forces the ExPKFI to use the ideal channel to re-initialize and so increase its performances independently of the measurement noise. Solutions that handle saturations could improve performances up to the ExPKFI curves.
3) For small measurement noise, MSE/R falls from 0.85 to 0.98 when the dynamic noise increases. The ExPKF seems to be more sensible to saturation effect for low channel noise.

In the next section we use a particle filter to handle correctly saturation effects so as to reach the ExPKFI performances.

IV. Particle Filter

Particle Filter (PF) was introduced by Gordon et al. in [14]. The same model 1 is extended to any nonlinear function f and h. Dynamic and measurement noise η_k ν_k are independent random variables with probability density function (pdf) $P_\eta(Q)$ and $P_\nu(R)$. That means the model is no more limited to the scope of polynomial function and can also represent non Gaussian noise as the saturation effects.

PF considers any pdf of the noise and propagate this noise in the nonlinear function. As equations are intractable in this general form a discrete approximation of the pdf is used, then the pdf of the random state $P(x_k)$ is approximated by a finite number N of particle ζ_k^i, $i = 1 : N$

$$P(x) \approx \sum_{i=1}^{N} w_k^i \delta_{\zeta_k^i}(x_k) \tag{22}$$

A. Particle filter equations

At each step time the particles ζ_k^i are propagated through 1 and the weights w_k^i are updated using Bayesian rules with the measurement.

Particles are very similar to the UKF sigma points. The main difference is that UKF use a minimal number of sigma points so as to estimate the Gaussian pdf propagation through the nonlinearity. A great number of particles is used to make a discrete approximation of the propagated pdf which can be of any form.

Starting from the prior pdf $P(x_0)$ of x , the posterior pdf is iteratively obtained using the following prediction 23 and update 24 relations:

$$P(x_k/Y_{k-1}) = \int_{x_k \in X} P(x_k/x_{k-1}) P(x_{k-1}/Y_{k-1}) dx_{k-1} \tag{23}$$

$$P(x_k/Y_k) = \frac{1}{c} P(y_k/x_k) \, P(x_k/Y_{k-1}) \tag{24}$$

where c is a normalizing constant and Y_{k-1} is the sequence $\{y_0, ..., y_{k-1}\}$ of available measurements.

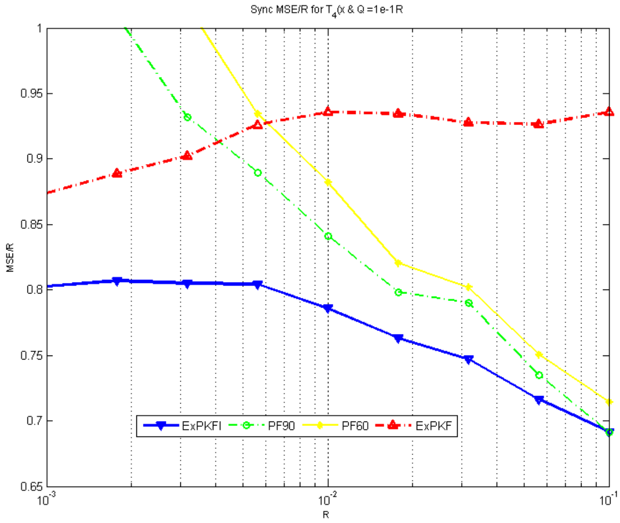

Fig. 3. Performance MSE/R for Fast ExPKF, ExPKFI and particle filter with 60 and 90 particles. Conditions are the same than for Fig.7 of Luca et al. paper including the relation $Q = \frac{R}{10}$

Particle filter keeps a representation of $P\left(x_k/_{Y_k}\right)$ with a set of weighed particles $\left\{\zeta_k^i, w_k^i\right\}$ with $(i = 1, ..., N)$ and $\left(\sum_{i=1}^{N} w_k^i = 1\right)$. The particle filter algorithm is initialized by[3]

$$\left\{\zeta_0^i, w_0^i\right\} iid{\sim} P(x_0)$$

The prediction equation 23 is approached by the distribution $P(x_k/Y_{k-1}) iid{\sim} \left\{\zeta_{k\_}^i, w_{k-1}^i\right\}$ of the propagated particles $\zeta_{k\_}^i$ where $\zeta_{k\_}^i = f(\zeta_{k-1}^i) + \eta_k^i$ with $\eta_k^i iid{\sim} P_\eta(Q)$

The update equation 24 is approached by the distribution $P(x_k/Y_k) iid{\sim} \left\{\zeta_k^i, w_k^i\right\}$ where $\zeta_k^i = \zeta_{k\_}^i$

$$w_k^i = \frac{w_{k-1}^i \, P(y_k/x_k = \zeta_k^i)}{\sum_{i=1}^{N} w_{k-1}^i \, P(y_k/x_k = \zeta_{k-1}^i)} \tag{25}$$

Finally the estimated state is the mean

$$\hat{x}_k = \sum_{i=1}^{N} w_k^i \zeta_k^i$$

The main problem of this algorithm is that the weight w_k^i are decreasing and that the filter is likely to degenerate. However an indicator for the degree of degeneracy:

$$N_k^{eff} = \frac{1}{\sum_{i=1}^{N} (w_k^i)^2} \tag{26}$$

which is the effective sample size presented in [15]. When $\frac{N_k^{eff}}{N} < 0.75$ it is necessary to resample the particle as proposed in the *generic particle filter* GPF (Algorithm 3 of [15]).

B. GPF applied to polynomial map

To compare with results obtained with Kalman filter, we apply the GPF to the same polynom $T_4(x)$ with Gaussian

[3] $iid{\sim}$ means identically independent distributed

noises η_k and ν_k. To handle saturations phenomenons, the noise η_k is replaced in the GPF filter by its Non-Gaussian uncorrelated counterpart $\eta_k^\star$. In fact this is simply done in the same way as for the emitter by computing Gaussian noise η_k and truncate the value of $f(x_k) + \eta_k$ into the set U.

It appears clearly in Fig 3 that the GPF works very efficiently for high channel noise and tends to the performance of the ExPKFI.

Small measurement noise lower the value of $P(y_k/x_k = \zeta_k^i)$ and then decrease rapidly the particle weights. Then several resampling steps are operated which explain the unexpected bad performances of the filter for low noise channel.

V. Conclusion

This paper offers two main results. First we redraw equations of the Exact Polynomial Kalman Filter. We minimize the Algorithm complexity to $O(N^2)$, however, we obtained a faster ExPKF algorithm which confirms the results of Luca et al. Then we apply particle filtering to polynomial chaotic maps to handle properly saturations that are drawn in the emitter to keep it stability. Performance in terms of mean square synchronization error are plotted and compared to previous results. The particle filter outperform all results and offer possibility to use synchronization under noisy channels for communication applications.

References

[1] M. B. Luca, S. Azou, G. Burel, and A. Serbanescu, *On Exact Kalman Filtering of Polynomial Systems*, IEEE Trans. Circuits Syst. I, vol. 53, no. 6. pp. 1329-1340, 2006.

[2] H. Fujisaka and T. Yamada, *Stability Theory of Synchronized Motion in Coupled-Oscillator Systems*, Prog. Theor. Phys., vol.69, pp. 32-47, 1983.

[3] L. Pecora and T. Caroll, *Synchronization in chaotic systems*, Phys. Rev. Lett., vol. 64, no. 2, pp. 821-823, 1990.

[4] M. Hasler *Synchronization of chaotic systems and transmission of information*, Int. J. Bifurcation and Chaos, vol. 8, no. 4, pp. 647-659, 1998.

[5] K. M. Cuomo, A. V. Oppenheim and S. H. Strogratz, *Synchronization of Lorenz-based chaotic circuits with application to communication*, IEEE Trans. Circuits Syst. II, vol. 40, no. 10, pp. 626-633, 1993.

[6] G. Kolumbán, M. P. Kennedy, and L. O. Chua, *The role of synchronization in digital communication using chaos — Part I: Fundamentals od digital communications*, IEEE Trans. on Circuits Syst. I vol 44, pp927–936, Oct 1997

[7] —- *Part II: Chaotic modulation and chaotic synchronisation*, IEEE Trans. on Circuits Syst. I vol 45, pp 1129–1140, Nov 1998

[8] —- *Part III: Performance bounds for correlation receivers*, IEEE Trans. on Circuits Syst. I vol 47, pp1673–1683, Dec 2000

[9] A. Gelb, *Applied Optimal Estimation*, MIT Press, Cambridge, 1974.

[10] Y. Bar-Shalom and X.-R. Li, *Estimation and Tracking: Principles, Techniques and Software*. Artech House, Boston, 1993.

[11] S. Julier, J. Uhlmann and H. F. Durrant-Whyte, *A new method for the nonlinear transformation of means and covariances in filters and estimators*, IEEE Trans. Automat. Contr., vol. 45, no. 3, pp. 477-482, 2000.

[12] E. A. Wan and R. van der Merwe, *Kalman Filtering and Neural Networks, chap. 7 : The Unscented Kalman Filter*, published by Wiley Publishing (editors S. Haykin), 2001.

[13] M. Norgaard, N. K. Poulsen and O. Ravn, *New developments in state estimation for nonlinear systems*, Automatica, vol. 36, pp. 1627-1638, 2000.

[14] N. J. Gordon, D. J. Salmond and A. F. M. Smith, *Novel approach to nonlinear/NonGuassian Bayesian State Estimation*, IEE Proc. vol. 140 no. 2, pp107-113, 1993

[15] M. S. Arulampalam, S. Maskell, N. Gordon and T. Clapp, *Tutorial on particle filter for online Nonlinear/NonGaussian Bayesian Tracking* IEEE trans. Signal Processing, vol. 50, no. 2, pp 1174–188, Feb. 2002