ADAPTIVE CONTROL OF PARAMETRIC STRICT FEEDBACK SYSTEMS WITH IMPROVED PERFORMANCE USING MODIFIED BACKSTEPPING

Erick Vile Grinits, Celso Pascoli Bottura

DMCSI – FEEC – UNICAMP
CP 6101, 13081-970, Campinas, SP, Brazil
E-mails: {egrinits, bottura}@dmcsi.fee.unicamp.br

Abstract: It is proposed a modification of the traditional adaptive backstepping method which leads to less control effort in the problem of non-linear control. The technique, which is applicable to parametric strict feedback systems, is built on a recently introduced Invariance Principle Extension and incorporates the use of optimisation techniques based on evolutionary computation to adjust the controller parameters. Simulations with the Chua's system are conducted to show the feasibility and effectiveness of the approach.

Keywords: Nonlinear control, adaptive control, optimization problems, genetic algorithms, chaotic behaviour.

1. INTRODUCTION

Over the last decade, the non-linear control field has experienced an impressive progress towards the development of successful methods aimed at constructing suitable control laws for complex non-linear systems. The most powerful of these techniques is the adaptive backstepping method (Krstic *et al.*, 1995), with which the issues of stabilization and tracking for several classes of non-linear systems with unknown parameters were able to be addressed to in a systematic fashion. In some applications, however, a backstepping-based control may feature an excessive control effort. It is of interest the design of general schemes of non-linear control which could maintain the systematisation of backstepping and simultaneously incorporate optimisation mechanisms so as to reduce the control effort.

According to the "No Free Lunch Theorem" (Wolpert & Macready, 1997), there is no general-purpose universal optimisation strategy. Classic methods and dedicated techniques outperform less conventional methodologies, like evolutionary algorithms (EAs) (Fleming & Purshouse, 2002), when restrictive hypotheses – such as continuity, differentiability, convexity, unimodality, etc. – on the search space are valid. On the other hand, EAs can deal with problems to which a detailed description is either too costly or not possible, or even about which it is not possible to assume such strong restrictions. Genetic Algorithms (GAs) (Michalewicz, 1996), in particular, have proved to be successful in problems that are difficult to formalize mathematically, such as optimised adaptive non-linear control with a discontinuous, non-differentiable, non-convex and/or multimodal search space (Fleming & Purshouse, 2002).

In view of this, it would be appropriate the incorporation of a GA in a non-linear control scheme built on systematic backstepping for a better performance as far as the control effort is concerned. Nevertheless, whereas the control law obtained with backstepping is Lyapunov-based it would be in principle useless the introduction of a GA to optimise the parameters of this controller, as the Invariance Principle requirements on which the backstepping technique is based pose excessive restrictions on the parameters search space (i.e. the parameters must comply with the non-negativeness demand for the Lyapunov function derivative).

However, extensions to classic stability requirements have been proposed. Rodrigues *et al.* (2000), for instance, advanced a generalization of the La Salle's Invariance Principle that includes the case in which the Lyapunov function derivative along the system solutions may be positive on a bounded set of the state space. Based on the new premises allowed by this Invariance Principle Extension (IPE), the traditional backstepping procedure can be modified so as to make its stability conditions less severe, thus enlarging the feasible region of the parameters search space and allowing the incorporation of a GA in order to obtain a set of parameters which may lead to a more efficient controller in terms of control effort.

An extension to the method of (Grinits & Bottura, 2004) (optimised control of a third order system using a modified backstepping procedure built on the IPE in conjunction with a GA) to a *general* class of strict feedback systems is presented here. By not requiring that the derivative of the Lyapunov functions should be nonpositive everywhere in the state space, the proposed methodology allows the combination of backstepping and GAs. As a result, the controller obtained may lead to a more efficient control process in terms of the control effort than when the traditional adaptive backstepping is used.

This paper is organized as follows. In Section 2, the IPE is reviewed. In Section 3, the new modified backstepping is presented. The Chua's system is used as an example to illustrate the feasibility and the advantages of the proposed approach in Section 4. Finally, conclusions are presented in Section 5.

2. THE INVARIANCE PRINCIPLE EXTENSION

In this section the IPE is reviewed. Its proof can be found in (Rodrigues, *et al.*, 2000). Consider the following autonomous differential equation (with $x \in \Re^n$):

$$\dot{x} = f(x), \quad x(0) = x_0 . \tag{1}$$

Theorem. Let $V : \Re^n \to \Re$ and $f : \Re^n \to \Re^n$ be C^1 functions. Let $C := \{x \in \Re^n : \dot{V}(x) > 0\}$. Suppose that $l := \sup_{x \in C} V(x) \in \Re$ and that $\overline{\Omega}_l := \{x \in \Re^n : V(x) \le l\}$ is bounded. Let $E := \{x \in \Re^n : \dot{V}(x) = 0\} \cup \overline{\Omega}_l$ and let B be the largest invariant set contained in E. Then, every solution $\varphi(t, x_0)$ of (1) that is bounded for $t \ge 0$ converges to the invariant set B as $t \to \infty$. Moreover, if $x_0 \in \overline{\Omega}_l$, then $\varphi(t, x_0)$ exists for all $t \ge 0$, $\varphi(t, x_0) \in \overline{\Omega}_l$ for all $t \ge 0$ and $\varphi(t, x_0)$ converges to the largest invariant set of (1) contained in $\overline{\Omega}_l$.

If it is assumed that $V : \Re^n \to \Re$ is radially unbounded, that is, if $V(x) \to \infty$ as $\|x\| \to \infty$, then every solution of (1) is bounded for $t \ge 0$ and the conclusions of the theorem hold for all solutions.

3. STABILISATION OF PARAMETRIC STRICT FEEDBACK SYSTEMS WITH MODIFIED BACKSTEPPING

The modified adaptive backstepping uses the above-reviewed IPE as a basis for the design of control laws that provide stability and convergence for a non-linear system without requiring the negative (semi)definiteness of the Lyapunov function derivative along its solutions everywhere in the state space. As mentioned before, this feature can lead to a more efficient control process in terms of control effort.

Consider the following n^{th} order uncertain non-linear system:

$$\begin{aligned}
\dot{x}_1 &= k_1 x_2 + \theta^T F_1(x_1, t) + f_1(x_1, t) \\
\dot{x}_2 &= k_2 x_3 + \theta^T F_2(x_1, x_2, t) + f_2(x_1, x_2, t) \\
&\vdots \\
\dot{x}_i &= k_i x_{i+1} + \theta^T F_i(x_1, \ldots, x_i, t) + f_i(x_1, \ldots, x_i, t) \\
&\vdots \\
\dot{x}_n &= g(\boldsymbol{x})u + \theta^T F_n(\boldsymbol{x}, t) + f_n(\boldsymbol{x}, t).
\end{aligned} \tag{2}$$

The system (2) is in parametric strict-feedback form, where $\boldsymbol{x} = [x_1 \ \cdots \ x_n]^T \in \Re^n$ is the state, $u \in \Re$ is the system input, $g(.) \ne 0$, $F_i(.)$ and $f_i(.)$, $i = 1, \ldots, n$, are known smooth non-linear functions and k_i, $i = 1, \ldots, n-1$, are nonzero scalars. It is supposed that $y = x_1 \in \Re$ is the system output. The objective is the tracking of the output y to a given set-point y_s, which is the output ($y_s = x_{r1}(t)$) of the reference model

$$\begin{aligned}
\dot{x}_{ri} &= f_{ri}(\boldsymbol{x_r}, t), \ 1 \le i \le m, \ n \le m \\
y_s &= x_{r1}(t),
\end{aligned} \tag{3}$$

where $\boldsymbol{x_r} = [x_{r1} \ \cdots \ x_{rm}]^T \in \Re^m$ is the state, y_s is the output and $f_{ri}(.)$, $i = 1, \ldots, m$, are known smooth non-linear functions.

The backstepping design procedure comprises n steps. At each step, an intermediate virtual control law is constructed using a quadratic Lyapunov function. As previously mentioned, the sense of the expression "Lyapunov function" in this paper includes the case in which its derivative may also be positive.

Step 1. Firstly, the auxiliary variable (error variable) corresponding to the system output is defined:

$$z_1 := x_1 - y_s , \tag{4}$$

where y_s is the desired set-point. Differentiating (4) along (2) and (3):

$$\dot{z}_1 = k_1(x_2 - x_{r2}) + k_1 x_{r2} + \theta^T F_1 + f_1 - f_{r1}. \tag{5}$$

The error variable corresponding to the second state variable is given by: $z_2 := x_2 - x_{r2} - \alpha_1$, where α_1 is the intermediate control law when $x_2 - x_{r2}$ is taken as a virtual control input. Thus:

$$\dot{z}_1 = k_1 z_2 + k_1 \alpha_1 + \theta^T F_{1s} + f_{1s}, \tag{6}$$

where $F_{1s} = F_1$ and $f_{1s} = f_1 - f_{r1} + k_1 x_{r2}$. The Lyapunov function associated with the subsystem (6) is introduced:

$$V_1 = \frac{1}{2}(z_1 - a_1)^2 + \frac{1}{2}\left(\theta - \hat{\theta}_{1st}\right)^T \Gamma^{-1}\left(\theta - \hat{\theta}_{1st}\right), \quad (7)$$

where $\Gamma = \Gamma^T > 0$ is the adaptive gain matrix, $a_1 \in \Re$ and $\hat{\theta}_{1st}$ is the parameter estimate vector for this step. The derivative of (7) along (6) is:

$$\dot{V}_1 = z_1\left(k_1 z_2 + k_1\alpha_1 + \theta^T F_{1s} + f_{1s}\right) - a_1\dot{z}_1 - \\ - \left(\theta - \hat{\theta}_{1st}\right)^T \Gamma^{-1}\dot{\hat{\theta}}_{1st}. \quad (8)$$

The intermediate control law α_1 is defined as:

$$\alpha_1 = \frac{1}{k_1}\left(-c_{11}z_1 - \hat{\theta}_{1st}^T F_{1s} - f_{1s}\right), \quad (9)$$

where $c_{11} \in \Re$ is a constant scalar. The z_1 subsystem may now be expressed as:

$$\dot{z}_1 = -c_{11}z_1 + k_1 z_2 + \left(\theta - \hat{\theta}_{1st}\right)^T F_{1s}. \quad (10)$$

With (9), (10) and update law $\dot{\hat{\theta}}_{1st} = \Gamma F_{1s}(z_1 - a_1)$:

$$\dot{V}_1 = -c_{11}z_1^2 + k_1 z_1 z_2 + a_1 c_{11} z_1 - a_1 k_1 z_2. \quad (11)$$

It is important to note that, contrary to what is required in the traditional backstepping procedure, negative values for c_{11} are allowed.

Step i ($2 \le i \le n-1$). The error variable corresponding to the i^{th} state variable is given by:

$$z_i := x_i - x_{ri} - \alpha_{i-1}. \quad (12)$$

Its derivative is

$$\dot{z}_i = k_i\left(x_{i+1} - x_{r,i+1}\right) + \theta^T F_{is} + f_{is}, \quad (13)$$

where

$$F_{is} = F_i - \sum_{j=1}^{i-1}\frac{\partial\alpha_{i-1}}{\partial x_j}F_j \text{ and}$$

$$f_{is} = f_i - f_{ri} + k_i x_{r,i+1} - \sum_{j=1}^{i-1}\frac{\partial\alpha_{i-1}}{\partial x_j}\left(k_j x_{j+1} + f_j\right) - \\ - \sum_{j=1}^{m}\frac{\partial\alpha_{i-1}}{\partial x_{rj}}f_{rj} - \sum_{j=1}^{i-1}\frac{\partial\alpha_{i-1}}{\partial\hat{\theta}_{jth}}\dot{\hat{\theta}}_{jth} - \frac{\partial\alpha_{i-1}}{\partial t}.$$

The error variable corresponding to the $(i+1)th$ state variable is given by: $z_{i+1} := x_{i+1} - x_{r,i+1} - \alpha_i$, where α_i is the intermediate control law when $x_{i+1} - x_{r,i+1}$ is taken as a virtual control input. We have then:

$$\dot{z}_i = k_i z_{i+1} + k_i\alpha_i + \theta^T F_{is} + f_{is}. \quad (14)$$

The intermediate Lyapunov function is introduced:

$$V_i = V_{i-1} + \frac{1}{2}(z_i - a_i)^2 + \frac{1}{2}\left(\theta - \hat{\theta}_{ith}\right)^T \Gamma^{-1}\left(\theta - \hat{\theta}_{ith}\right) (15)$$

where $a_i \in \Re$ and $\hat{\theta}_{ith}$ is the parameter estimate vector for this step. The derivative of (15) along the previous z_j subsystems ($j = 1, ..., i-1$) and (14) is

$$\dot{V}_i = -\sum_{j=1}^{i-1}\sum_{t=1}^{i-1}c_{j,t}z_j z_t - \left(\theta - \hat{\theta}_{ith}\right)^T \Gamma^{-1}\dot{\hat{\theta}}_{ith} - a_i\dot{z}_i \overset{(i>2)}{+} \\ \overset{(i>2)}{+} \sum_{j=1}^{i-2}\left(\sum_{t=j}^{i-1}a_t c_{j,t} + a_{j+1}k_j - a_{j-1}k_{j-1}\right)z_j + \\ + \left(a_{i-1}c_{i-1,i-1} \overset{(i>2)}{-} a_{i-2}k_{i-2}\right)z_{i-1} - a_{i-1}k_{i-1}z_i + \\ + z_i\left(k_{i-1}z_{i-1} + k_i z_{i+1} + k_i\alpha_i + \theta^T F_{is} + f_{is}\right).$$

The intermediate control law α_i is defined as:

$$\alpha_i = \frac{1}{k_i}\left(-k_{i-1}z_{i-1} - \sum_{j=1}^{i}c_{j,i}z_j - \hat{\theta}_{ith}^T F_{is} - f_{is}\right) (16)$$

where $c_{j,i} \in \Re$, $j = 1, ..., i$, are constant scalars. The z_i subsystem may now be expressed as

$$\dot{z}_i = \overset{(i>2)}{-}\sum_{j=1}^{i-2}c_{j,i}z_j - \left(c_{i-1,i} + k_{i-1}\right)z_{i-1} - \\ - c_{i,i}z_i + k_i z_{i+1} + \left(\theta - \hat{\theta}_{ith}\right)^T F_{is}. \quad (17)$$

With (16), (17) and update law

$$\dot{\hat{\theta}}_{ith} = \Gamma F_{is}(z_i - a_i) \quad (18)$$

we get

$$\dot{V}_i = -\sum_{j=1}^{i}\sum_{t=1}^{i}c_{j,t}z_j z_t + k_i z_i z_{i+1} + \\ + \sum_{j=1}^{i-1}\left(\sum_{t=j}^{i}a_t c_{j,t} + a_{j+1}k_j - a_{j-1}k_{j-1}\right)z_j + \\ + \left(a_i c_{i,i} - a_{i-1}k_{i-1}\right)z_i - a_i k_i z_{i+1}. \quad (19)$$

Step n ($n \ge 2$). The auxiliary variable corresponding to the last state variable is given by:

$$z_n := x_n - x_{rn} - \alpha_{n-1} \quad (20)$$

and its derivative is

$$\dot{z}_n = gu + \theta^T F_{ns} + f_{ns}, \quad (21)$$

where

$$F_{ns} = F_n - \sum_{j=1}^{n-1}\frac{\partial\alpha_{n-1}}{\partial x_j}F_j \text{ and}$$

$$f_{ns} = f_n - f_{rn} - \sum_{j=1}^{n-1}\frac{\partial\alpha_{n-1}}{\partial x_j}\left(k_j x_{j+1} + f_j\right) - \\ - \sum_{j=1}^{m}\frac{\partial\alpha_{n-1}}{\partial x_{rj}}f_{rj} - \sum_{j=1}^{n-1}\frac{\partial\alpha_{n-1}}{\partial\hat{\theta}_{jth}}\dot{\hat{\theta}}_{jth} - \frac{\partial\alpha_{n-1}}{\partial t}.$$

The Lyapunov function of the whole system is:

$$V_n = V_{n-1} + \frac{(z_n - a_n)^2}{2} + \frac{\left(\theta - \hat{\theta}_{nth}\right)^T \Gamma^{-1}\left(\theta - \hat{\theta}_{nth}\right)}{2} (22)$$

where $a_n \in \Re$ and $\hat{\theta}_{nth}$ is the parameter estimate vector for this last step. The derivative of (22) along the previous z_j subsystems (j = 1, …, n-1) and (21) is

$$\dot{V}_n = -\sum_{j=1}^{n-1}\sum_{t=1}^{n-1}c_{j,t}z_j z_t - \left(\theta - \hat{\theta}_{nth}\right)^T \Gamma^{-1}\dot{\hat{\theta}}_{nth} - a_n\dot{z}_n \overset{(n>2)}{+} \\ \overset{(n>2)}{+} \sum_{j=1}^{n-2}\left(\sum_{t=j}^{n-1}a_t c_{j,t} + a_{j+1}k_j - a_{j-1}k_{j-1}\right)z_j + \\ + \left(a_{n-1}c_{n-1,n-1} \overset{(n>2)}{-} a_{n-2}k_{n-2}\right)z_{n-1} - a_{n-1}k_{n-1}z_n + \\ + z_n\left(k_{n-1}z_{n-1} + gu + \theta^T F_{ns} + f_{ns}\right).$$

The controller can now be chosen as

$$u = \frac{1}{g}\left(-k_{n-1}z_{n-1} - \sum_{j=1}^{n}c_{j,n}z_j - \hat{\theta}_{nth}^T F_{ns} - f_{ns}\right), (23)$$

where $c_{j,n} \in \Re$, $j = 1, ..., n$, are constant scalars. The $\dot{z}_n$ subsystem (21) can now be expressed in its definitive form:

$$\dot{z}_n = \overset{(n>2)}{-} \sum_{j=1}^{n-2} c_{j,n} z_j - \left(c_{n-1,n} + k_{n-1}\right) z_{n-1} - \\ - c_{n,n} z_n + \left(\theta - \hat{\theta}_{nth}\right)^T F_{ns}. \quad (24)$$

With the update law

$$\dot{\hat{\theta}}_{nth} = \Gamma F_{ns}\left(z_n - a_n\right) \quad (25)$$

and substituting (23) for u in $\dot{V}_n$, we get the expression for the Lyapunov function derivative of the whole system:

$$\dot{V}_n = -\sum_{j=1}^{n}\sum_{t=1}^{n} c_{j,t} z_j z_t \overset{(n>2)}{+} \\ \overset{(n>2)}{+} \sum_{j=1}^{n-2}\left(\sum_{t=j}^{n} a_t c_{j,t} + a_{j+1}k_j - a_{j-1}k_{j-1}\right) z_j + \\ + \left(a_{n-1}c_{n-1,n-1} + a_n c_{n-1,n} + a_n k_{n-1} \overset{(n>2)}{-} a_{n-2}k_{n-2}\right) z_{n-1} + \\ + \left(a_n c_{n,n} - a_{n-1}k_{n-1}\right) z_n.$$

Following the notation adopted in Section 2, the set C is defined as

$$C := \left\{ z \in \Re^n : \dot{V}_n > 0 \right\} \quad (26)$$

According to the IPE, the adaptive control of the strict feedback system (2) will be achieved if the set C is bounded, as the Lyapunov function (22) is radially unbounded. The equation $\dot{V}_n = 0$ represents several kinds of geometric loci. Depending on the values assumed by the set of parameters $\{c_{j,t} \in \Re : j = 1,\ldots,n; t = j,\ldots,n\}$ the set C will be bounded or unbounded. There are geometric tests based on the coefficients of the quadratic form $\dot{V}_n$ which can be conducted to assess the boundedness of C.

Both the shape of C and the performance of the controller as far as the control effort is concerned rely on the values assumed by the set of parameters $\{c_{j,t} \in \Re : j = 1,\ldots,n; t = j,\ldots,n\}$. This allows us to apply optimisation techniques on these parameters in order to achieve a more efficient control process. As the constraint represented by the boundedness of C is a requisite for stability, we get the following optimisation task:

$$\{c_{i,j}\} = \arg\min_{\{c_{i,j}\}} \left[\begin{pmatrix} \text{magnitude} \\ \text{of } u \end{pmatrix} + \begin{pmatrix} \text{transient duration} \\ \text{of state variables} \\ \text{of } \boldsymbol{x} \end{pmatrix} \right] \quad (27) \\ \text{subject to} \begin{Bmatrix} \text{boundedness of set } C \text{ determined by} \\ \text{the coefficients of } \dot{V}_n \end{Bmatrix}$$

where i, j = 1, …, n. (27) aims not only at low control effort magnitudes, but also at an acceptable transient duration. Appropriate formalisation and description associated with conventional optimisation methods (e.g. based on gradient) are not obtainable for (27). Indeed, the objective-function of (27) does not allow the calculation of derivatives and search space characteristics whose knowledge is necessary to the application of those methods (continuity, convexity, etc.) are not verifiable. Optimisation techniques based on evolutionary computation are therefore more adequate to the optimisation task (27).

It is important to notice that with the traditional backstepping (Krstic, *et al.*, 1995) such an optimisation task is not possible, because, as already mentioned, the Lyapunov Direct Method and the La Salle's Invariance Principle requirements on which the backstepping technique is based pose excessive restrictions on the parameters search space.

4. AN EXAMPLE – CONTROL OF UNCERTAIN CHUA'S SYSTEM WITH IMPROVED PERFORMANCE

Analog electronic circuits are well-known examples of systems exhibiting non-linear response. Among these systems, the Chua's circuit has become a paradigm, due to its simplicity and richness of behaviours. We work with the Chua's circuit in its dimensionless form:

$$\begin{aligned} \dot{x}_1' &= p_1 x_2' - p_2 x_1' - p_3\left(|x_1'+1| - |x_1'-1|\right) \\ \dot{x}_2' &= p_4 x_1' - p_5 x_2' + p_6 x_3' \\ \dot{x}_3' &= -p_7 x_2'. \end{aligned} \quad (28)$$

The equations (28) can be rendered into parametric strict feedback form with the following state variables transformations: $x_1 = x_3'$, $x_2 = x_2'$ and $x_3 = x_1'$. So, with $b_1 = p_7$, $b_2 = p_4$, $\theta_1 = p_6$, $\theta_2 = p_5$, $\theta_3 = p_1$, $\theta_4 = p_2$ and $\theta_5 = p_3$ we get:

$$\begin{aligned} \dot{x}_1 &= -b_1 x_2 \\ \dot{x}_2 &= b_2 x_3 + \theta_1 x_1 - \theta_2 x_2 \\ \dot{x}_3 &= u + \theta_3 x_2 - \theta_4 x_3 - \theta_5\left(|x_3+1| - |x_3-1|\right), \end{aligned} \quad (29)$$

where a controller $u(.)$ is assumed to be fed into the third equation in (29). In comparison with the strict feedback form (2) and in the case when the parameters $\theta = [\theta_1, \theta_2, \ldots, \theta_5]^T$ are unknown, we have

$$k_1 = -b_1, k_2 = b_2, g(x) = 1, f_1(.) = f_2(.) = f_3(.) \equiv 0 \\ F_1(.) = [0 \quad 0 \quad 0 \quad 0 \quad 0]^T, F_2(.) = [x_1 \quad -x_2 \quad 0 \quad 0 \quad 0]^T, \\ F_3(.) = [0 \quad 0 \quad x_2 \quad -x_3 \quad -(|x_3+1| - |x_3-1|)]^T.$$

Our aim is the design of an adaptive state-feedback controller which guarantees regulation of the state $\boldsymbol{x} = [x_1 \quad x_2 \quad x_3]^T$ at the origin and boundedness of all the signals (state variables, control, parameter estimates) in the closed-loop system with as less control effort as possible.

Following the steps presented in Section 3 with $a_1 = a_2 = a_3 = 1$, we arrive at the control law expression

$$u = -b_2 z_2 - c_{13}z_1 - c_{23}z_2 - c_{33}z_3 + x_1\dot{\hat{\theta}}_2^{(1)} - x_2\dot{\hat{\theta}}_2^{(2)} - \\ - \left[(b_1/b_2).\left(b_1 - c_{12} - \hat{\theta}_2^{(1)}\right)\right] x_2 + \left(c_{11} + c_{22} - \hat{\theta}_2^{(2)}\right) x_3 - \\ - \left[\left(c_{11} + c_{22} - \hat{\theta}_2^{(2)}\right)/b_2\right]\left(x_1\hat{\theta}_3^{(1)} - x_2\hat{\theta}_3^{(2)}\right) - \\ - x_2\hat{\theta}_3^{(3)} + x_3\hat{\theta}_3^{(4)} - \left(|x_3+1| - |x_3-1|\right)\hat{\theta}_3^{(5)}, \quad (30)$$

where $c_{11}, c_{22}, c_{33}, c_{12}, c_{13}, c_{23} \in \Re$ are constant scalars and $\hat{\theta}_i = \left[\hat{\theta}_i^{(1)} \quad \cdots \quad \hat{\theta}_i^{(5)}\right]^T$, $i = 1,2,3,$ are the parameter estimates; at the update laws

- $\dot{\hat{\theta}}_1^{(1)} = \dot{\hat{\theta}}_1^{(2)} = \dot{\hat{\theta}}_1^{(3)} = \dot{\hat{\theta}}_1^{(4)} = \dot{\hat{\theta}}_1^{(5)} = 0;$
- $\dot{\hat{\theta}}_2^{(1)} = \Gamma_1 x_1(z_2 - 1), \dot{\hat{\theta}}_2^{(2)} = -\Gamma_2 x_2(z_2 - 1),$
  $\dot{\hat{\theta}}_2^{(3)} = \dot{\hat{\theta}}_2^{(4)} = \dot{\hat{\theta}}_2^{(5)} = 0;$
- $\dot{\hat{\theta}}_3^{(1)} = \Gamma_1 \dfrac{c_{11} + c_{22} - \hat{\theta}_2^{(2)}}{b_2} x_1(z_3 - 1),$ (31)

$$\dot{\hat{\theta}}_3^{(2)} = -\Gamma_2 \frac{c_{11} + c_{22} - \hat{\theta}_2^{(2)}}{b_2} x_2(z_3 - 1),$$

$$\dot{\hat{\theta}}_3^{(3)} = \Gamma_3 x_2(z_3 - 1),\ \dot{\hat{\theta}}_3^{(4)} = -\Gamma_4 x_3(z_3 - 1),$$

$$\dot{\hat{\theta}}_3^{(5)} = \Gamma_5(|x_3 + 1| - |x_3 - 1|)(z_3 - 1);$$

and at the expression for the Lyapunov function derivative of the whole system:

$$\begin{aligned}\dot{V} = &-c_{11}z_1^2 - c_{22}z_2^2 - c_{33}z_3^2 - c_{12}z_1z_2 - \\ &- c_{13}z_1z_3 - c_{23}z_2z_3 + (c_{33} - b_2)z_3 \\ &+ (c_{11} + c_{12} + c_{13} - b_1)z_1 + \\ &+ (c_{22} + c_{23} + b_1 + b_2)z_2.\end{aligned} \tag{32}$$

According to the notation, the set C is defined as

$$C := \{z \in \Re^3 : \dot{V} > 0\} \tag{33}$$

As mentioned in the preceding section, the shape of the set C and the performance of the controller (30) with update laws (31) will rely on the values assumed by the set $\{c_{11}, c_{22}, c_{33}, c_{12}, c_{13}, c_{23}\}$.

We employ a genetic algorithm (GA) (Michalewicz, 1996) in order to determine a satisfactory set of parameters for the control law (30) and update laws (31). The GA is used off-line to search through a population of controllers (i.e. through a population of sets of parameters $\{c_{11}, c_{22}, c_{33}, c_{12}, c_{13}, c_{23}\}$) the member most fit to be implemented. It is important to point out that, as previously stated, in this problem (see (27)) it is not possible to use neither classic nor dedicated optimisation methods.

One of the requisites for the GA to find the best solution to a particular problem is that the individuals of the population must be encoded into a form upon which the GA can operate efficiently. Here the population has 100 chromosomes and each one has six genes – thus correlating with the set of six controller parameters –, whose alleles can take any value in the range [-10, 10] with a precision of four digits after the decimal point. Considering the general case, the proposed method can be easily extended to n^{th} order systems. In that case the GA features a chromosome with $n^2 - \sum_{k=1}^{n-1} k$ genes.

GA processes include biomimetic operations such as selection, crossover and mutation. Based on the idea that, on average, the members of the population of the current generation should be as good (or better) at maximising the fitness function than those of the previous generation, we utilise a variant of the *elitist strategy* in which the 20 fittest members survive, undisturbed, in the next generation. Crossover combines the features of two parent chromosomes to form two similar offspring. The offspring may then replace weaker individuals in the population. We employ the *arithmetical crossover* with the 80 fittest members being selected for reproduction; of these, parents are randomly chosen, with equal probability. We also utilise *non-uniform mutation*, which precludes the GA from converging to local solutions. The GA is run over 50 generations.

The fitness function takes into account the objective aimed at: control effort magnitudes as small as possible without an excessive enlargement of the transient response duration. An adequate transient duration should be no greater than $t_t = 2$ units of time. For each chromosome at the generation t, we carry out a simulation in order to evaluate its performance (i.e. the performance of the corresponding controller) in terms of the fitness function

$$fitness = \frac{1}{p + \int_0^2 u^2(\tau) + \beta.\|x(\tau)\|_2 \, d\tau}, \tag{34}$$

where the Euclidean norm $\|x(.)\|_2$ represents the effect of the transient and β is a weighting factor; here, $\beta = 100$. Before proceeding with the simulation, each individual is considered feasible or infeasible. The feasible individuals are the ones which make the C set bounded.

The boundedness of C is determined through the following matrices:

$$M_1 = \begin{bmatrix} -c_{11} & \frac{-c_{12}}{2} & \frac{-c_{13}}{2} \\ \frac{-c_{12}}{2} & -c_{22} & \frac{-c_{23}}{2} \\ \frac{-c_{13}}{2} & \frac{-c_{23}}{2} & -c_{33} \end{bmatrix}, M_2 = \left[\begin{array}{c|c} M_1 & \begin{matrix} M_2^{14} \\ M_2^{24} \\ M_2^{34} \end{matrix} \\ \hline \begin{matrix} M_2^{14} & M_2^{24} & M_2^{34} \end{matrix} & 0 \end{array}\right], \tag{35}$$

where $\begin{cases} M_2^{14} = 0.5(c_{11} + c_{12} + c_{13} - b_1), \\ M_2^{34} = 0.5(c_{33} - b_2), \\ M_2^{24} = 0.5(c_{22} + c_{23} + b_1 + b_2). \end{cases}$

The following conditions must be satisfied as far as the matrices (35) are concerned: rank(M_1) = 3, rank(M_2) = 4, det(M_2) < 0 and the real parts of the eigenvalues of M_1 must have the same sign. In this case the expression $\dot{V} = 0$ will correspond to an ellipsoid. The infeasible chromosomes are penalised with $p = 10^{10}$. Simulations are conducted only for the feasible individuals (in this case, $p = 0$).

The parameters of the Chua's system adopt the following values:

$$\begin{aligned} &b_1 = 16;\ b_2 = 1;\ \theta_1 = 1;\ \theta_2 = 1; \\ &\theta_3 = 9.8008;\ \theta_4 = 2.8028;\ \theta_5 = -2.1021. \end{aligned} \tag{36}$$

For these values the Chua's system (29) exhibit a chaotic response when $u = 0$. The initial conditions are $x_1(0) = 0.2$, $x_2(0) = 0.5$ and $x_3(0) = 0.3$. After

50 generations, the best chromosome consists of the following genes:

$$c_{11} = -1.4056\,;\, c_{22} = -6.7052\,;\, c_{33} = -0.3747\,; \quad c_{12} = 4.9612\,;\, c_{13} = 0.5991;\, c_{23} = -0.1743. \tag{37}$$

The boundary of set C corresponding to these parameters is an ellipsoid. Since C is a convex set and the Lyapunov function (22) with $n = 3$ is a convex function, the $\sup_{z \in C} V(z)$ occurs at the boundary of the set C. The set $\overline{\Omega}_l$ is a sphere and the set C is contained in $\overline{\Omega}_l$. So, every solution converges to the largest invariant set contained in $\overline{\Omega}_l$. The introduction of the control law (30) and update laws (31) with the parameters (37) and $\Gamma = I$ (the identity matrix) into the uncertain Chua's system makes the z state trajectory converge to the origin $z = 0$. In view of this and as far as the expressions of the error variables are concerned, the x state trajectory also converges to the origin $x = 0$, thereby achieving the regulation objective.

The Fig. 1 shows the time responses of the state variables x_1, x_2 and x_3 when the controller (30) and update laws (31) are applied to the system (29) (solid lines). It confirms the effectiveness of the design scheme with regard to the stabilisation objective with transient duration $t_t \approx 2$. It is also shown the time responses obtained when the traditional adaptive backstepping procedure (Krstic, *et al.*, 1995) with Lyapunov function

$$V = \tfrac{1}{2}\sum_{i=1}^{3} z_i^2 + \tfrac{1}{2}\sum_{i=1}^{3}\left(\theta - \hat{\theta}_{ith}\right)^T \Gamma^{-1}\left(\theta - \hat{\theta}_{ith}\right)$$

is applied (dotted lines). In that case the control law parameters are $c_{11} = c_{22} = c_{33} = 3$ and $c_{12} = c_{13} = c_{23} = 0$. We also choose $\Gamma = I$ for the sake of comparison. These are the parameter values which lead to $t_t \approx 2$. It is important to point out that in the traditional backstepping we must have $c_{11}, c_{22}, c_{33} > 0$ and $c_{12} = c_{13} = c_{23} = 0$.

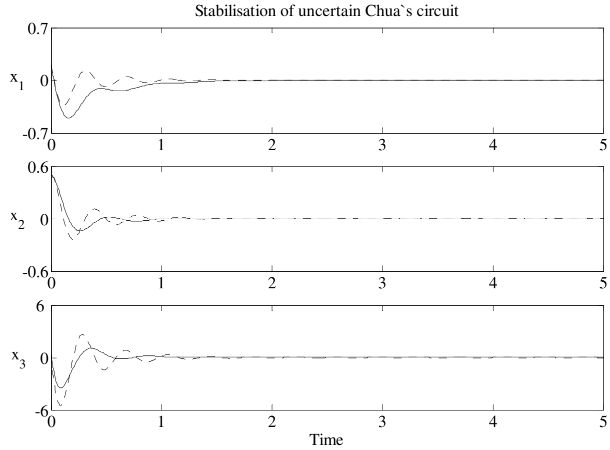

Fig. 1. Time response of the state variables.

The Fig. 2 shows the control effort required to the adaptive regulation objective when we apply the controller (30) and update laws (31) into the uncertain Chua's system (solid line). It is also shown the control effort when it is applied the controller built on the traditional adaptive backstepping procedure (dotted line). The control effort magnitude is reduced when the controller derived from the modified adaptive backstepping design with parameters optimised via GA is applied. It is important to note that there are no values for the parameters $c_{11}, c_{22}, c_{33} > 0$ of the traditional adaptive backstepping controller that lead to less control effort with $t_t \approx 2$.

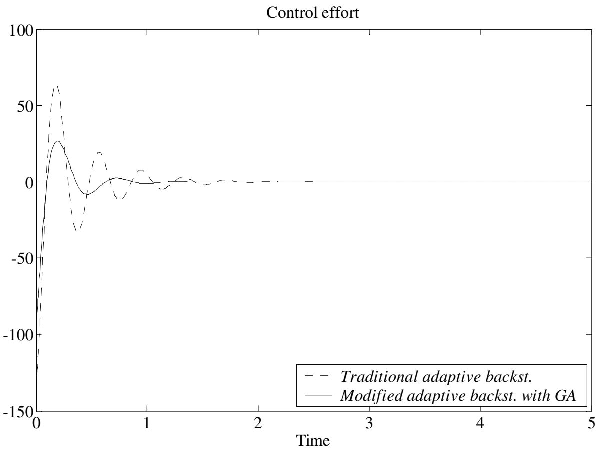

Fig. 2. Control effort magnitude reduction.

5. CONCLUSION

It is proposed a modification of the traditional adaptive backstepping grounded on a recent Extension to the Invariance Principle that allows the incorporation of optimisation methods based on evolutionary computation and can lead to a more efficient performance in the control of parametric strict feedback systems as far as the control effort is concerned than when the traditional backstepping is applied. The Chua's circuit was used as an example of the effectiveness of the approach.

REFERENCES

Fleming, P.J. and R. C. Purshouse (2002). Evolutionary algorithms in control systems engineering: a survey. *Control Engineering Practice,* **10(11),** 1223-1241.

Grinits, E.V. and C.P. Bottura (2004). Control of Uncertain Chaotic Systems with Improved Performance Using Modified Adaptive Backstepping and Genetic Algorithms. *Proceedings of the 2004 IEEE International Symposium on Intelligent Control.* Taiwan, 466-471.

Krstic, M., I. Kanellakopoulos and P. Kokotovic (1995). *Nonlinear and adaptive control design*. New York, John Wiley.

Michalewicz, Z. (1996). *Genetic algorithms + data structures = evolution programs*. Berlin, Springer.

Rodrigues, H., L. Alberto and N. Bretas (2000). On the invariance principle: generalizations and applications to synchronization. *IEEE Trans. on Circuits and Syst., I,* **47(5)**, 730-739.

Wolpert, D.H. and W.G. Macready (1997). No free lunch theorems for optimization. *IEEE Trans. on Evolutionary Computation*, **1(1)**, 67-82.