# VARIABLE STRUCTURE CONTROL DESIGN BASED ON DIFFERENTIAL EVOLUTION OPTIMIZATION WITH NEW SINOSOIDAL GAUSSIAN OPERATOR

**Leandro dos Santos Coelho**
Pontifical Catholic University of Paraná, LAS/PPGEPS/CCET/PUCPR, Production and Systems Engineering Graduate Program
Imaculada Conceição, 1155, Zip code 80215-901, Curitiba, Paraná, Brazil
leandro.coelho@pucpr.br

**Renato Krohling**
Lehrstuhl Elektrische Steuerung und Regelung
Fakultät für Elektrotechnik und Informationstechnik
sd Universität Dortmund, D-44221 Dortmund, Germany
krohling@est.e-technik.uni-dortmund.de

***Abstract.*** *This paper considers the non-linear control of industrial processes based on self-tuning discrete variable structure control (VSC) technique. A novel and systematic VSC design methodology is proposed, which integrates an estimator based on recursive least square algorithm, a discrete quasi-sliding surface and a optimization method. Contrary to the trial and error selection of the variable structure feeedback gains reported in the literature, the selection in present work is done using differential evolution (DE) optimization with a new sinusoidal Gaussian operator.The proposed design has been applied to a control valve described by a non-linear Wiener model. Numerical simulation results reveal that the new DE approach is applicable and promising for the VSC design.*

***Keywords***: *variable structure control, optimization, differential evolution.*

## 1. Introduction

Advanced conceptions to design industrial control systems are, in general, dependent of mathematical models of the controlled process. Also, the task of the controllers is to achieve an optimum performance when facing with various types of disturbances that are, sometimes, unknown in practical applications. An advanced methodology for control design is the structure variable control (VSC) with sliding mode. Practically all design methods for variable structure control systems are based on concepts of sliding modes.

VSC is a structure that adjusts and controls based on the deviation of the system states apart from the sliding surface. By using sliding mode control strategies, the designer needs to select an appropriate sliding surface to make the system's movement coincident with this sliding surface through discontinuous control behavior (Yu *et al*., 2004). VSC with its high robustness to parametric uncertainties and exogenous disturbances, has become a very attractive control method in recent years (Li and Wikander, 2004).

Since the VSC introduced in 1977, many useful approaches for sliding surface have been proposed, such as eigenstructure assignment method (Elghezawi *et al*., 1983), Lyapunov approach (Su *et al*., 1996), fuzzy sliding surface (Lee *et al*., 1998), optimization based on computational intelligence (Kaynak and Rudas, 1998), reference model (Li and Wikander, 2004), internal model approach (Camacho *et al*., 2003), PID (proportional-integral-derivative) control (Li *et al*., 2001), adaptive control (Cho and Edge, 2000) and others. The VSC has been successfully applied to robot manipulators, underwater vehicles, automotive transmissions and engines, chemical processes, high-performance electric motors, and power systems (Hung *et al*., 1993).

The self-tuning design philosophy (Wellstead and Zarrop, 1991) is an effective procedure to be applied to control systems whose model parameters are unknown and uncertain. However, if the plant identification task is not completed, the self-tuning control gives undesirable responses. The sliding-mode control based on variable structure systems is robust to deal with small uncertainty in the plant. When the uncertainty is larger than one taken from design specifications, the sliding mode provides unstable control signal and undesirable responses can be obtained. The implementation of the VSC by computer control requires discrete model and brings undesirable responses due to chattering phenomenon. In addition, the self-tuning approach in the VSC design is implemented by recursive least-squares algorithm (RLS), as shown in Furuta *et al.* (1989), Furuta (1993) and Lee and Oh (1998).

This paper presents a self-tuning discrete-time VSC design using an optimization method based on differential evolution (DE) optimization — an evolutionary computation approach — with a new sinusoidal Gaussian operator. The proposed VSC algorithm prevents the fluctuation of the estimated parameters that occur during the parameter adaptation and ensures stable closed-loop response. The proposed design has been applied to a control valve described by a non-linear Wiener model. Numerical simulation results reveal that the new DE approach is applicable and promising for the VSC design.

The paper is organized as follows. Basic concepts of self-tuning discrete VSC are presented in section 2. In section 3, the design procedure of the VSC based on differential evolution is derived. In the section 4, the simulation results are presented and discussed. The conclusions and futures works are commented in the section 5.

## 2. Variable structure control systems

In the formulation of any control problem there will typically be discrepancies between the real plant and the mathematical model developed for controller design purposes. This plant model mismatch may be due to unknown dynamics, variation in system parameters or the approximation of a complex plant behavior by a straightforward model. The engineer must ensure that the control algorithm had the ability to find the required performance levels in spite of plant model mismatches. So, in the process control field there is a great interest in the development of robust control methodologies to solve this problem. One particular approach to design a robust control controller is the so-called sliding mode control method.

The theory of VSC was developed for robust control of uncertain nonlinear systems. Continuous-time VSC has been extensively studied and has been used in various applications. Much less is known of discrete-time sliding mode controllers. In practice it is often assumed that the sampling frequency is sufficiently high to assume that the closed-loop system is continuous-time. Another possibility is to design the sliding mode controller in discrete-time, based on a discrete-time model of the sampled system under control.

The VSC design consists of two components. The first involves the design of a switching function so that the sliding motion satisfies design specifications. The second is concerned with the selection of a control law that will make the switching function attractive to the system state. Note that this control law is not necessarily discontinuous.

In sliding mode control, the VSC is designed to drive and then constrain the system state to lie within a neighborhood of the switching function. There are two main advantages to this approach. First, the dynamic behavior of the system may be tailored by the particular choice of switching function. Second, the closed-loop response becomes totally insensitive to a particular class of uncertainty. The latter invariance property clearly makes the methodology an appropriate candidate for robust control. In addition, the ability to specify performance directly makes sliding mode control attractive from the design viewpoint. In the next section a self-tuning discrete VSC approach is described.

### 2.1 Self-tuning discrete VSC

Discrete sliding mode control (quasi-sliding mode control) has received attention recently. These controllers are formulated of numerous ways for the systems with different kinds uncertainties using switching or non-switching types of techniques. The existence of sliding mode cannot be guaranteed in the presence of uncertainties in the discrete-time systems. The characteristics of discrete-time variable structure control systems differ from those of continuous-time variable structure systems in two aspects. First, the discrete-time variable structure systems can only undergo quasi-sliding modes. Second, when the state does reach the switching surface, the subsequent discrete-time switching cannot generate the equivalent control to keep the state on the surface (Hung *et al.*, 1993).

This section presents discrete type variable structure control (quasi-sliding mode control) design proposed by Furuta *et al.* (1989), Furuta (1993) and Lee and Oh (1998) and its application to adaptive control with estimated parameters by RLS algorithm.

In proposed approach the quasi-sliding mode control system uses Lamarckian evolution for design parameters optimization of control law. The Lamarckian evolution for design configuration of discrete sliding mode control system provides an automatic and optimized selection of sliding slopes and controller parameters, offering superior performances in transient, steady-state and robustness to manual designs.

In this case, the Furuta's design is utilized in second order processes with disturbances and non-modeled dynamics. The mathematical model of process is represented by equation:

$$A(q^{-1})y(t) = q^{-d}B(q^{-1})u(t) \tag{1}$$

where $u(t)$ is the input and $y(t)$ is the output. Polynomials $A(q^{-1})$ and $B(q^{-1})$ have no common terms and $q^{-1}$ denotes the time shift operator defined by $q^{-1}y_k = y_{k-1}$. $A(q^{-1})$ and $B(q^{-1})$ are assumed known and given by

$$A(q^{-1}) = 1 + a_1q^{-1} + a_2q^{-2} + \ldots + a_nq^{-na}$$
$$B(q^{-1}) = b_o + b_1q^{-1} + b_2q^{-2} + \ldots + b_mq^{-nb}$$

The RLS algorithm is utilized to update estimates of parameters $\theta=\{a_1, a_2, b_0, b_1\}$ of the ARMA model of second order. The definition of the sliding hypersurfaces is given by:

$$s(t+1) = e(t+1) + k_1 e(t) + k_2 e(t-1) = 0 \tag{2}$$

and the error is given by equation

$$e(t) = y(t) - y_r(t) \tag{3}$$

where $k_1$ and $k_2$ are determined so that the error is stable on the hypersurface . The control signal is chosen to be in form of

$$\begin{bmatrix} e(t) \\ e(t+1) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k_2 & -k_1 \end{bmatrix} \begin{bmatrix} e(t-1) \\ e(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} v'(t) \tag{4}$$

$$v'(t) = e(t) + k_1 e(t-1) + k_2 e(t-2) + \begin{bmatrix} f_1 & f_2 \end{bmatrix} \begin{bmatrix} e(t-1) \\ e(t) \end{bmatrix} \tag{5}$$

where the last component $\begin{bmatrix} f_1 & f_2 \end{bmatrix} \begin{bmatrix} e(t-1) \\ e(t) \end{bmatrix}$ is the switching term of the equation, and the component $e(t) + k_1 e(t-1) + k_2 e(t-2)$ driving the state along the sliding hypersurfaces. A relevant definition is a positive definite function $V(t)$ given by

$$V(t) = \frac{1}{2} s(t)^2 \tag{6}$$

and the definition of $\Delta s(t+1)$ as the difference

$$\Delta s(t+1) = s(t+1) - s(t) \tag{7}$$

From equation (7) is obtained the following equation:

$$V(t+1) = V(t) + 2s(t)\Delta s(t+1) + \Delta s(t+1)^2 \tag{8}$$

The control objective will be to make to decrease along the switching hypersurfaces. From equation (8), the following condition is obtained:

$$s(t)\Delta s(t+1) < -\frac{1}{2}\left[\Delta s(t+1)\right]^2 \tag{9}$$

Using the equation (7), the control signal has the following equations:

$$\Delta s(t+1) = \begin{bmatrix} f_1 & f_2 \end{bmatrix} \begin{bmatrix} e(t-1) \\ e(t) \end{bmatrix} \tag{10}$$

$$\Delta s(t+1)s(t) = f_1 e(t-1)s(t) + f_2 e(t)s(t) < -\delta_1 f_0 - \delta_2 f_0 < -\frac{1}{2}\left[ f_0^2 e(t-1)^2 + 2 f_0^2 e(t-1)e(t) + f_0^2 e(t)^2 \right] \tag{11}$$

where

$$\delta_i = \frac{1}{2}\left[ f_0 e(t+i-2) + f_0 |e(t-1)| |e(t)| \right], \qquad f_0 > 0, i = 1,2 \tag{12}$$

$$f_i = \begin{cases} f_0 & \text{if } e(t+i\text{-}2)s(t) < -\delta \\ 0 & \text{if } e(t+i\text{-}2)s(t) < \delta, \\ -f_0 & \text{if } e(t+i\text{-}2)s(t) > \delta \end{cases} \qquad f_0 > 0,\ i = 1,2 \tag{13}$$

The cost function $J$ that will optimized is

$$J = p\left[y(t+1) - v(t) - v'(t)\right]^2 + r\left[u(t) - u(t-1)^2\right] \tag{14}$$

The following control law makes the cost function (14) minimal. Using the estimated parameters the above control law becomes:

$$\begin{aligned} u(t) = \frac{1}{\hat{b}_1 + r}\{ & y_r(t+1) - \hat{a}_1 y(t) - \hat{a}_2 y(t-1) - [\hat{b}_2 - r]u(t-1) - [1-k_1]e(t) \\ & -[k_1 - k_2]e(t-1) - k_2 e(t-2) - \left[f_1 \quad f_2\right]\begin{bmatrix} e(t) \\ e(t-1) \end{bmatrix} \end{aligned} \tag{15}$$

### 3. Optimization procedure using differential evolution

DE is a population-based and stochastic function minimizer (or maximizer), whose simple, yet powerful, and straightforward features make it very attractive for numerical optimization. DE uses a rather greedy and less stochastic approach to problem solving compared to EAs. DE combines simple arithmetic operators with the classical operators of crossover, mutation and selection to evolve from a randomly generated starting population to a final solution.

Storn and Price (1995) first introduced the DE algorithm a few years ago. The DE was successfully applied to the optimization of some well-known non-linear, non-differentiable and non-convex functions in Storn (1997). DE is an approach for the treatment of real-valued optimization problems. In this case, Krink *et al*. (2004) mentioned also that DE is a very powerful heuristic for non-noisy optimization problems, but that noise is indeed a serious problem for conventional DE, when the fitness of candidate solutions approaches the fitness variance caused by the noise.

DE is similar to a ($\mu$, $\lambda$) evolution strategies, but in DE the mutation is not done via some separately defined probability density function. DE is also characterized by the use of a population-derived noise to adapt the mutation rate of the evolution process, implementation simplicity and speed of operation.

There are two variants of DE that have been reported, DE/*rand*/1/*bin* and DE/*best*/2/*bin*. The different variants are classified using the following notation: DE/$\alpha$/$\beta$/$\delta$, where $\alpha$ indicates the method for selecting the parent chromosome that will form the base of the mutated vector, $\beta$ indicates the number of difference vectors used to perturb the base chromosome, and $\delta$ indicates the crossover mechanism used to create the child population. The *bin* acronym indicates that crossover is controlled by a series of independent binomial experiments.

The fundamental idea behind DE is a scheme by which it generates the trial parameter vectors. DE, at each time step, mutates vectors by adding weighted, random vector differentials to them. If the cost of the trial vector is better than that of the target, the target vector is replaced by trial vector in the next generation. The variant implemented in this paper was the DE/*rand*/1/*bin* and given by the following steps:

(i) Initialize a population of individuals (solution vectors) with random values generated according to a uniform probability distribution in the $n$ dimensional problem space.

(ii) For each individual, evaluate its fitness value.

(iii) Mutate individuals in according to equation:

$$\mathrm{z}_i(t+1) = \mathrm{x}_{i,r_1}(t) + f_m\left[\mathrm{x}_{i,r_2}(t) - \mathrm{x}_{i,r_3}(t)\right] \tag{16}$$

(iv) Following the mutation operation, crossover is applied in the population. For each mutant vector, $z_i(t+1)$, an index $rnbr(i) \in \{1, 2, \cdots, n\}$ is randomly chosen using uniform distribution, and a *trial vector*, $u_i(t+1) = \left[u_{i_1}(t+1), u_{i_2}(t+1), \ldots, u_{i_n}(t+1)\right]^{\mathrm{T}}$, is generated with

$$u_{i_j}(t+1) = \begin{cases} z_{i_j}(t+1), & \text{if } (randb(j) \leq CR) \text{ or } (j = rnbr(i)), \\ x_{i_j}(t), & \text{if } (randb(j) > CR) \text{ or } (j \neq rnbr(i)) \end{cases} \tag{17}$$

To decide whether or not the vector $u_i(t+1)$ should be a member of the population comprising the next generation, it is compared to the corresponding vector $x_i(t)$. Thus, if $F_c$ denotes the objective function under minimization, then

$$x_i(t+1) = \begin{cases} u_i(t+1), & \text{if } F_c(t+1) < F_c(\mathrm{x}_i(t)), \\ x_i(t), & \text{otherwise} \end{cases} \tag{18}$$

(iv) Loop to step (ii) until a stopping criterion is met, usually a maximum number of iterations (generations), $G_{max}$.

In the above equations, $i$ =1,2,...,$N$ is the individual's index of population; $j$ =1,2,..., $n$ is the position in $n$ dimensional individual; $t$ is the time (generation); $x_i(t) = \left[x_{i_1}(t), x_{i_2}(t), ..., x_{i_n}(t)\right]^T$ stands for the position of the $i$-th individual of population of $N$ real-valued $n$-dimensional vectors; $z_i(t) = \left[z_{i_1}(t), z_{i_2}(t), ..., z_{i_n}(t)\right]^T$ stands for the position of the $i$-th individual of a *mutant vector*; $r_1$, $r_2$ and $r_3$ are mutually different integers and also different from the running index, $i$, randomly selected with uniform distribution from the set $\{1, 2, \cdots, i-1, i+1, \cdots, N\}$; $f_m > 0$ is a real parameter, called *mutation factor*, which controls the amplification of the difference between two individuals so as to avoid search stagnation and it is usually taken from the range [0.1, 1]; *randb(j)* is the $j$-th evaluation of a uniform random number generation with [0, 1]; $CR$ is a *crossover rate* in the range [0, 1]; and $F_c$ is the evaluation of cost function. Usually, the performance of a DE algorithm depends on three variables: the population size $N$, the mutation factor $f_m$, and the crossover rate $CR$..

### 3.1. New differential evolution approaches

The choice of mutation factor, $f_m$, affects the performance of DE. In this context, the new approaches of DE based on sinosoidal and Gaussian functions for the $f_m$ setup are described as follows (see figures 1 to 3):

*Approach 1* – DE(1): The parameter $f_m$ of equation (16) is modified by the formula (19) through the following equation based on a sinosoidal function:

$$z_i(t+1) = x_{i,r_1}(t) + \left[\alpha + A\sin(\omega\beta)\right]\left[x_{i,r_2}(t) - x_{i,r_3}(t)\right] \tag{19}$$

where $\alpha + A\sin(\omega\beta)$ represents $f_m$; $\alpha$ is a DC component of signal, $A$ is the amplitude of signal, ω is the angular frequency of signal, $\beta$ is a gain. The choice of values was $\alpha$ =0.4, $A$=0.2, ω=0.1$G_{max}$ ($G_{max}$ is the maximum number of generations of optimization procedure) and $\beta$ are values linearly spaced between the initial value -180 degree and the final value of 180 degree with increments based on $G_{max}$.

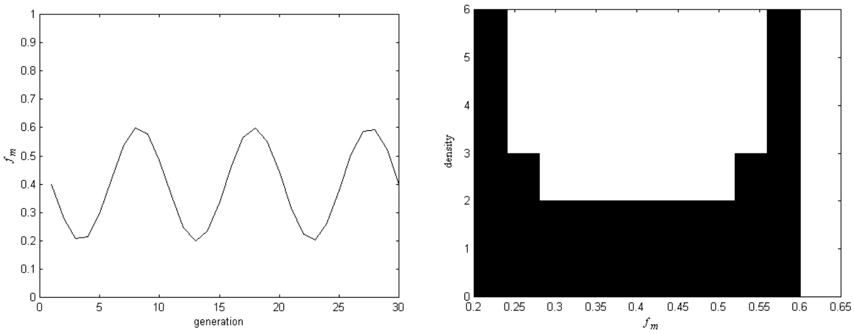
*Approach 2* – DE(2): The parameter $f_m$ of equation (16) is modified by the formula (20) through the following equation based on random numbers with Gaussian distribution:

$$z_i(t+1) = x_{i,r_1}(t) + \left[\alpha + 0.1Gauss(\cdot)\right]\left[x_{i,r_2}(t) - x_{i,r_3}(t)\right] \tag{20}$$

where $\alpha + 0.1Gauss(\cdot)$ represents $f_m$ of equation (16); $\alpha$ is a DC component of signal and $Gauss(\cdot)$ generates random numbers using Gaussian distribution with zero mean.

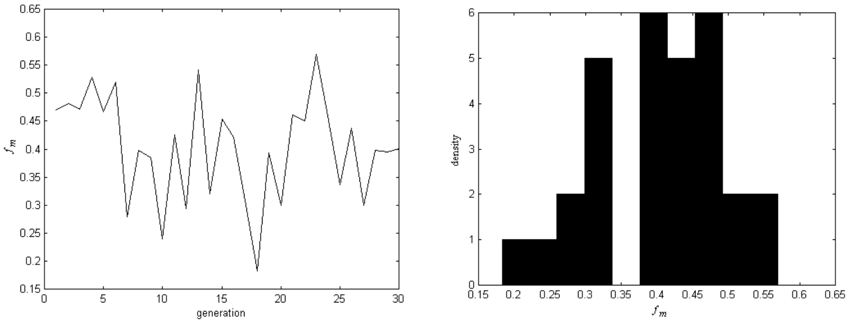
*Approach 2* – DE(3): This approach is a combination of DE(1) and DE(2) approaches. The parameter $f_m$ of equation (16) is modified by the formula (21) through the following equation:

$$z_i(t+1) = x_{i,r_1}(t) + \left\{\alpha + \left[A\sin(\omega\beta)\right]\left[0.1Gauss(\cdot)\right]\right\}\left[x_{i,r_2}(t) - x_{i,r_3}(t)\right] \tag{21}$$



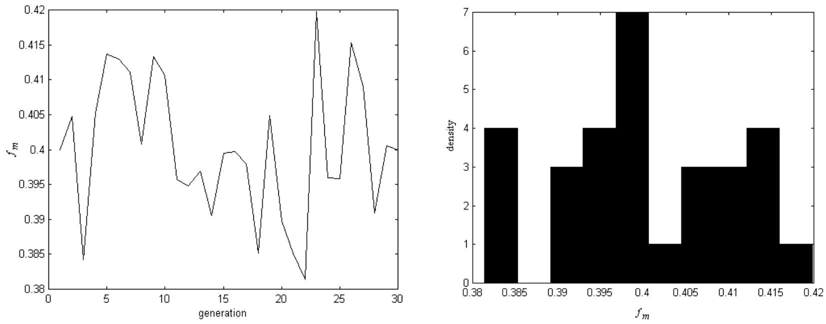

(a) (b)

Figure 1. (a) $f_m$; (b) Orbital points' distribution of $f_m$ of DE(1).




(a) (b)

Figure 2. (a) $f_m$; (b) Orbital points' distribution of $f_m$ of DE(2).




(a) (b)

Figure 3. (a) $f_m$; (b) Orbital points' distribution of $f_m$ of DE(3).

### 3.2. VSC design based on differential evolution

The proposed VSC design presents two stages. In the first, the identification of the nonlinear process by least mean squares algorithm is realized. In the second, the optimization of VSC design parameters is considered. The design procedure uses an off-line configuration to get the process model and the model validation is realized on-line with the process.

To implement discrete-time VSC based on based on DE optimization, three important problems are considered. They are: choice of dynamic sliding surface for the process, computation of the discrete-time dynamic sliding surface variable, and self-tuning of the switching control magnitude to reduce chattering. The optimization procedure of VSC parameters by DE is presented in figure 4.

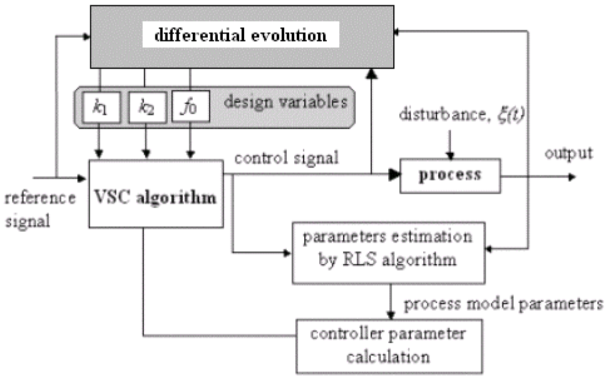



Figure 4. Optimization procedure of VSC design by differential evolution.

## 4. Case study and analysis of results

### 4.1. Description of case study (control valve)

The control valve system is an opening with adjustable area. Normally it consists of an actuator, a valve body and a valve plug. The actuator is a device that transforms the control signal to movement of the stem and valve plug. Wigren (1993) describes the plant where the control valve dynamic is described by a Wiener model (the nonlinear element follows linear block) and it is given by

$$x(t) = 1{,}5714x(t-1) + 0{,}6873x(t-2) + 0{,}0616u(t\text{-}1) + 0{,}0543u(t\text{-}2) \tag{22}$$

$$y(t) = f_n(x(t)) = \frac{x(t)}{\sqrt{0{,}10 + 0{,}90[x(t)]^2}} \tag{23}$$

where $u(t)$ is the control pressure, $x(t)$ is the stem position, and $y(t)$ is the flow through the valve which is the controlled variable. The nonlinear behavior of the control valve described by equation (23) is shown in figure 5. The input to the process, $u(t)$, is constrained between [0; 0.4].

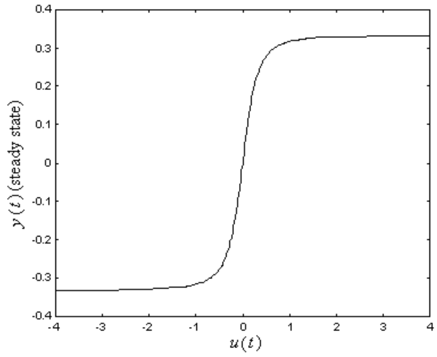



Figure 5. Static characteristic of a control valve.

### 4.2. Simulation results

The effectiveness and advantages of the proposed self-tuning VSC based on DE optimization is demonstrated through controlling a control valve, where.

- *Prediction model*: The prediction model uses $na$=$nb$=2. The parameters of polynomials $A(q^{-1})$ and $B(q^{-1})$ are estimated using a RLS algorithm. An exponential forgetting factor is adopted with values ranging from 0.95 to unity. The estimated parameters are initialized in simulation with $=\{a_1, a_2, b_0, b_1\}=\{0.2, 0.2, 0.2, 0.2\}$. The diagonal of the initial covariance matrix is set to 1000. The covariance matrix is reinitiated for each change of reference.
- *Reference trajectory*: The desired reference signal is given by $y_r(t)$=0.2 for the samples 1-100, $y_r(t)$=0.6 for the samples 101-200, and $y_r(t)$=0.8 for the samples 201-300.
- *Objective function*: The objective function of VSC is given by equation (14) and the control law is set to the equation (15).
- *Optimization procedure*: The DE is used in optimization procedure of parameters $k_1$, $k_2$ and $f_0$ of VSC. In the sequel it illustrates the main features of the DE employed:
  (i) *Fitness function*: The fitness function to be maximized is represented by equation, $fitness = k/(1 + J)$, where the scale coefficient is $k$=3. The performance index, J, is given by equation (14) with $p$=1 and $r$=0.01.
  (ii) *Constraint of control signal*: The constraint of control signal is $0 \le u(t) \le 1.2$.
  (iii) *Design parameters of* DE *approaches*: $CR$ = 0.8, population size, $N$ = 10, and $G_{max}$ = 30.

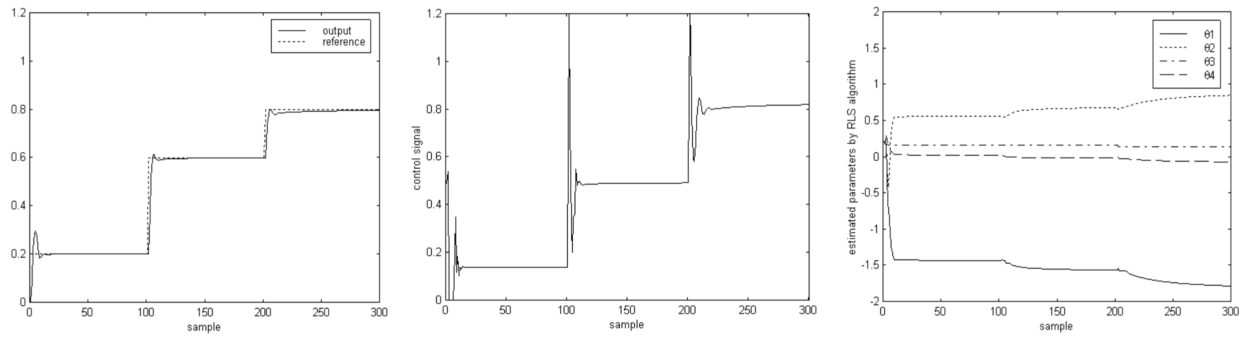
(iv) *Stopping criterion*: The adopted termination criterion is the number of generations is equal to 30. In this case, the number of experiments (runs) for each design is set to 10.

This section shows the results obtained with the VSC design. Table 1 summarizes the performance and design parameters of VSC optimized by DE approaches. Figure 6 shows the performance of best VSC design using DE(3). Servo and regulatory simulation results of the VSC using DE(3) for a control valve system are shown in Figure 6(a) and 6(b), respectively. The regulatory behavior is based on the rejection of additive disturbances in the process output when: (i) sample 70: $y(t) = y(t) + 0.2$; (ii) sample 150: $y(t) = y(t) - 0.4$; (iii) sample 260: $y(t) = y(t) - 0.4$; and (iv) sample 260: $y(t) = y(t) - 0.2$.

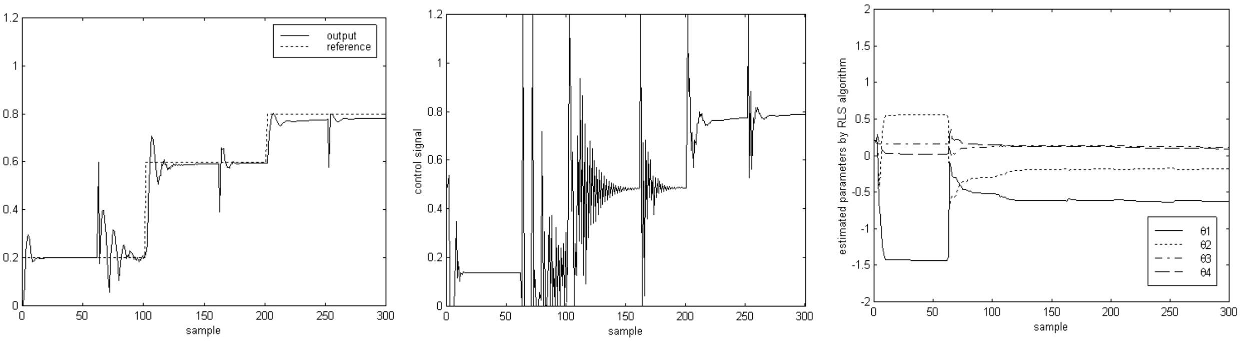
Table 1. Simulation results for self-tuning VSC (best fitness for 10 experiments).

| Differential evolution Approach | Best design based on $J$ | | | Fitness function | | | |
|---|---|---|---|---|---|---|---|
| | $k_1$ | $k_2$ | $f_0$ | best | average | minimum | standard deviation |
| Classical DE ($f_m = 0.4$) | 0.5230 | -0.0298 | 0.4488 | 0.7502 | 0.7393 | 0.7236 | 0.0113 |
| DE(1) | 0.4820 | -0.0472 | 0.4489 | 0.7515 | 0.7421 | 0.7194 | 0.0132 |
| DE(2) | 0.4998 | -0.0442 | 0.4492 | 0.7517 | 0.7403 | 0.7269 | 0.0117 |
| DE(3) | **0.5149** | **-0.0456** | **0.4510** | **0.7527** | **0.7474** | **0.7359** | **0.0067** |

Performance of VSC designs was affected by nonlinearity of control valve. Furthermore, the VSC designs obtained fast response, reasonable control activity, and good setpoint tracking ability. The good performance indicated by the VSC using DE approaches confirms the usefulness and robustness of the proposed method for practical applications.

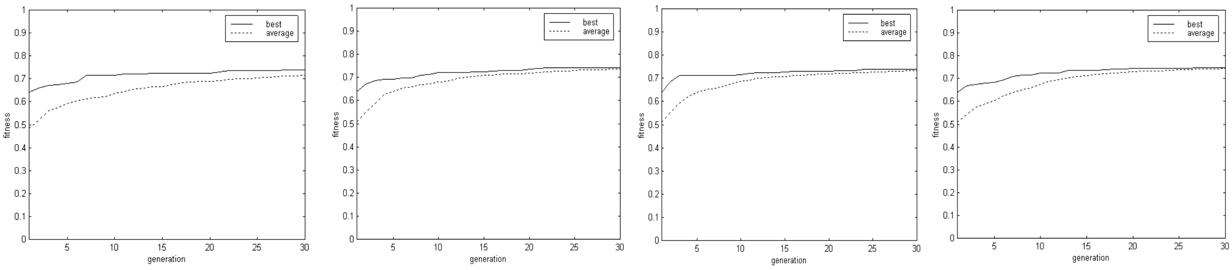



(a) tracking different reference signals (optimization phase)




(b) rejection of perturbations (validation phase)

Figure 6. Best results of VSC design based on DE(3).




(a) DE(1) (b) DE(2) (c) DE(3) (d) DE(4)

Figure 7. Average convergence of fitness of best individual of population for DE approaches (10 runs).

## 5. Conclusion and future works

This paper presented the development of a Lamarckian evolution method to the VSC design. In the paper, the effectiveness of the proposed control schemes were shown in simulations of a CSTR process. The utilization of Lamarckian approach avoids the tedious manual trial-and-error procedure and it presents robustness in tuning of VSC design parameters.

The aim of future works is to investigate the use of Lamarckian evolution combined with other computational intelligence methodologies, such as fuzzy systems and neural networks for optimization in multivariable processes.

Furthermore, others relevant studies will can be realized, such as: (i) comparative analysis of several approaches of Lamarckian evolution based on simulated annealing and simplex method, (ii) verification of number of floating-point operations of each design, and (iii) design conceptions for constrained problems.

## 6. References

Camacho, O., Smith, O. and Moreno, W., 2003, "Development of an Internal Model Sliding Mode Controller", Ind. Eng. Chem. Res.,Vol. 42, pp. 568-573.

Cho, S. H. and Edge, K. A., 2000, "Adaptive Sliding Mode Tracking Control of Hydraulic Servosystems with Unknown Non-Linear Friction and Modelling Error", Proc. Instin. Mech. Engrs, Part I, IMechE, Vol. 214, pp. 247-257.

Elghezawi, O. M. E., Zinober, A. S. I. and Billings, S. A., 1983, "Analysis and Designed of Variable Structure Systems Using a Geometric Approach", International Journal of Control, Vol. 38, pp. 657-671.

Furuta, K., 1993, "VSS Type Self-Tuning Control", IEEE Transactions on Industrial Electronics, Vol. 40, No. 1, pp. 37-44.

Furuta, K., Kosuge, K. and Kobayashi, K., 1989, "VSS-Type Self-Tuning Control of Direct-Drive Motor", Proceedings of Annual Conference of the IEEE Industrial Electronics Society, Philadelphia, PA, pp. 281-286.

Hung, J. Y., Gao, W. and Hung, J. C., 1993, "Variable Structure Control: A Survey", IEEE Transactions on Industrial Electronics, Vol. 40, No. 1, pp. 2-22.

Kaynak, O. and Rudas, I. J., 1998, "The Fusion of Computational Intelligence Methodologies in Sliding Mode Control", Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society, Aachen, Germany, pp. T25-T34.

Krink, T., Filipic, B., Fogel, G.B. and Thomsen, R., 2004, "Noisy Optimization Problems - A Particular Challenge for Differential Evolution?", Proceedings of the 6th IEEE Congress on Evolutionary Computation, Portland, Vol. 1, pp. 332-339.

Lee, H., Kim, E., Kna, H. J. and Park, M., 1998, "Designed of Sliding Mode Controller with Fuzzy Sliding Surfaces", IEE Proceedings Control Theory Appl., Vol. 145, No. 5, pp. 411-418.

Lee, P. -M. and Oh, J. -H., 1998, "Improvements on VSS-Type Self-Tuning Control for a Tracking Controller", IEEE Transactions on Industrial Electronics, Vol. 45, No. 2, pp. 319-325.

Li, M., Wang, F. and Gao, F., 2001, "PID-Based Sliding Mode Controller for Nonlinear Processes", Ind. Eng. Chem. Res.,Vol. 40, pp. 2660-2667.

Li, Y. -F. and Wikander, J., 2004, "Model Reference Discrete-Time Sliding Model Control of Linear Motor Precision Servo Systems", Mechatronics, Vol. 14, pp. 835-851.

Storn, R. and Price, K., 1995, "Differential Evolution: A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces", Technical Report TR-95-012, International Computer Science Institute, Berkeley, USA.

Storn, R., 1997, "Differential Evolution — A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces", Journal of Global Optimization, Vol. 11, No. 4, pp. 341-359.

Su, W. C. Drakunov, S. V. and Özgüner, Ü., 1996, "Constructing Discontinuity Surfaces for Variable Structure System: a Lyapunov Approach", Automatica, Vol. 32, pp. 925-928.

Wellstead, P. E. and Zarrop, M. B., 1991, Self-Tuning Systems: Control and Signal Processing, Chichester: Wiley.

Wigren, T., 1993, "Recursive Prediction Error Identification Using the Nonlinear Wiener Model", Automatica, Vol. 29, No. 4, pp. 1011-1025.

Yu, W. -C., Wang, G. -J. and Chang, C. –C., 2004, "Discrete Sliding Mode Control with Forgetting Dynamic Sliding Surface", Mechatronics, Vol. 14, pp. 737-755.

## 7. Responsibility notice

The authors are the only responsible for the printed material included in this paper.