# Support Vector Machines Based Adaptive Controller for Piston Hydraulic Motor

Hossam Mohammad Khalil and Mohamad El Bardini

*Industrial Electronics & Control, Menofia University, 32952, Menouf, Egypt*
*Phone: 002-010-0244-110; Fax: 002-048-3660-716;*
*E-mail: Hossam_kha@yahoo.com.*

***Abstract***

*Speed control of hydraulic actuator is mainly desirable in diverse hydraulic systems including those for agriculture and most of them are for manufacturing. In all these systems the main demand is to accurately control the speed of the hydraulic actuator under variable operating conditions including various set-points and disturbances. In this paper a generalized predictive control (GPC) which is well known is used based on least squares support vector machines (LS-SVM) as a model reference control. The main drawback of LS-SVM is that the sparseness of standard SVM is lost which means that some of the system parameters are equal to zero. In this contribution a new method for sparseness is introduced. The electro-hydraulic servo system is modeled recursively each sample based on LS-SVM using linear kernel function avoiding linearization when radial basis function (RBF) is employed. Based on the model obtained the control signal is calculated at each sample using GPC. The presented controller performance is compared to that of self-tuning PID controller based on recursive least squares (RLS) algorithm.*

***Keywords:*** *Hydraulic Actuator; Generalized Predictive Controller (GPC); Least squares support vector machines (LS-SVM); Sparseness.*

## 1. Introduction

In traditional neural networks many decisions had to be made such as the number of neurons, the length of the training cycle and the existence of many local minima, which are eliminated here [1,2,3]. Support vector machines (SVM) [4,5] have an increasingly attention, as they introduce some useful prosperities, which make them preferred in solving a large number of regression problems. Their solutions are characterized by convex optimization problems, up to the determination of a few additional tuning parameters. Moreover, model complexity follows from this convex optimization problem. Typically, one solves a convex quadratic programming (QP) problem in order to determine the SVM model. The formulation of the optimization problem associated with this QP problem involves inequality constraints. An interesting property of the solution is that one obtains a sparse approximation, in the sense that many elements in the QP solution vector are equal to zero [6]. SVM is a kernel based approach, which allows the use of linear, polynomial and RBF kernels and others that satisfy Mercer's condition [7]. Least Squares SVM (LS-SVM) is a modified SVM version that uses equality constraints and a sum squared error (SSE) cost function to have a set of linear equations instead of solving a QP problem inferred by the standard SVM. This simplifies the computations and enhances the speed considerably. LS-SVM based algorithms have been developed very rapidly and have been applied to many areas [8,9]. However, despite these computationally attractive features, LS-SVM solution also has a potential drawback which is the loss of sparseness. In this case every data point is contributing to the model and the relative importance of a data point is given by its support value. The sparseness of standard SVM can also be reached with LS-SVM. This was achieved by weighted LS-SVM [9], fixed-size LS-SVM [10] and pruning [11].

The generalized predictive control (GPC) [12,13] is one of the most powerful and useful model reference control methods for a wide class of nonlinear systems [14,15,16,17,18,19]. The basic principle of the GPC is to generate a sequence of control signals at each sample which optimizes the control effort, in order to follow exactly the desired speed. The GPC, integrating self-adaptive control into predictive control, has been shown to be practically effective. Linear predictive control approaches are well-established in control practice, and nonlinear model predictive control (NMPC) [20,21] has also found its way in control practice. NMPC algorithms are based on various nonlinear models and its quality depends mainly on the quality of the model [22]. In [23,24], LS-SVM has been utilized to obtain the model of nonlinear plants with different nonlinearity in the GPC scheme but without taking into consideration the main drawback of modelling algorithm which is the loss of sparseness. Also in [25,26], GPC based on LS-SVM is used to control different nonlinear systems using RBF kernel which is then linearized to obtain the ARMA model that can be used by GPC. In this contribution the GPC paradigm used in [23] is used with linear kernel function with new pruning algorithm, in which the length of data set is set fixed, The oldest data pair is omitted and in the same time a new data pair is added which is not used before unlike older methods which take much more time to obtain a sparse solution.

Electro-hydraulic servo systems are found in diverse applications including manufacturing systems, active suspension systems, fatigue testing, flight simulation, paper machines, electromagnetic marine engineering, injection moulding machines, aluminium mill equipment, radar scanners, naval gunnery and robotics. Hydraulic actuators are also common in aircraft, where their high power-to-weight ratio and precise control makes them an ideal choice for actuation of flight surfaces [27]. However the dynamics of hydraulic systems are highly non-linear, the system may be subjected to non-smooth nonlinearities due to control input saturation, directional change of valve opening, friction, and valve overlap. Aside from the non-linear nature of the hydraulic dynamics, there are many considerable model nonlinearities, such as external disturbances and leakages that cannot be modelled exactly. Furthermore, the non-linear functions that describe the nonlinearities may not be known [28]. Hydraulic actuators do not only allow the generation of large forces, but are thanks to modern control technology and sensors, also capable of assuming control tasks such as closed-loop velocity controlling or highly precise positioning of heavy loads [29]. The hydraulic motor used in this context is of the piston type which has much nonlinearity such as friction problem (static and dynamic frictions) and movement instability, also the solenoid-valve is of the proportional flow type which has some nonlinearity such as dead-band and flow gain hysteresis which makes the system under control is highly nonlinear.

The motivation of this paper is to implement the GPC based on the new pruning algorithm of LS-SVM [24] as a model based speed controller for piston-type rotary hydraulic actuator. The GPC control signal is calculated continuously based on the identified LS-SVM model. The implemented controller can deal with nonlinear systems effectively such as the electro-hydraulic system used here as the modelling paradigm is effective for nonlinear systems. The implemented controller is then compared to the well known self-tuning PID controller based on RLS modelling algorithm.

The paper is organized as follows: The system is described in details in section II. The LS-SVM paradigm for modelling nonlinear systems is introduced in section III. In section IV, the online LS-SVM is given with the method for sparseness. The GPC algorithm based on an online LS-SVM using linear kernel is described in details in section V. The stability properties of an open loop GPC and its closed-loop scheme are introduced in section VI. The results for implementing the presented controller versus the self-tuning PID controller based on RLS for different operating point and load disturbances are introduced in section VII.

## 2. Hydraulic System Description

The overall electro-hydraulic system (EHS160) scheme involves:

a) Source of mechanical power (220 V A.C. Motor) driving a pump to produce a fixed supply of hydraulic fluid pressure which is 7MPa.

b) Device for converting hydraulic flow into mechanical one, the rotary hydraulic actuator is that device (piston hydraulic motor including 5 pistons).

c) Device to control the fluid flow which in turn controls the output quantity (speed), the electro-hydraulic solenoid-valve of proportional type is implemented.

d) Ancillary devices (accumulator - filter - pressure gauges- temperature gauge).

e) The other important item in the system is a mechanical load to receive the work output of the hydraulic actuator which in our case a mechanical load applied directly to the output shaft of the hydraulic motor through brake pads.

f) An I/O industrial interface card with 12-bits through which a real-time control system is built where the output of the system is connected to A/D input CH4 and the control signal is applied to the system through D/A output CH1.

The speed of the actuator is measured through a tachogenerator which is fed directly to the computer as a feedback signal. Within the computer, the feedback signal is employed to calculate the error which in turn produces the suitable control action to maintain the speed of the hydraulic motor at the desired one. The control signal calculated is fed to the system again through a D/A channel to control the actuator's speed output. The experimental set-up of the system is shown Fig.(1), where the schematic diagram of the system under study is shown in Fig.(2).

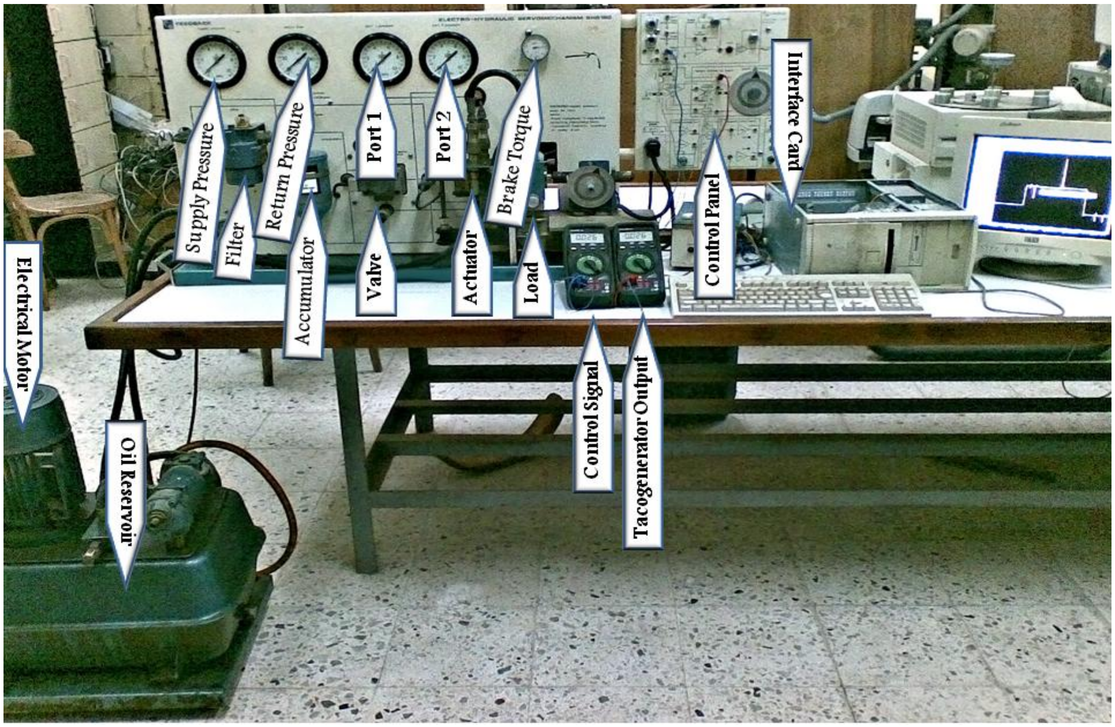


**Figure 1. The experimental setup of the overall real time hydraulic control system.**

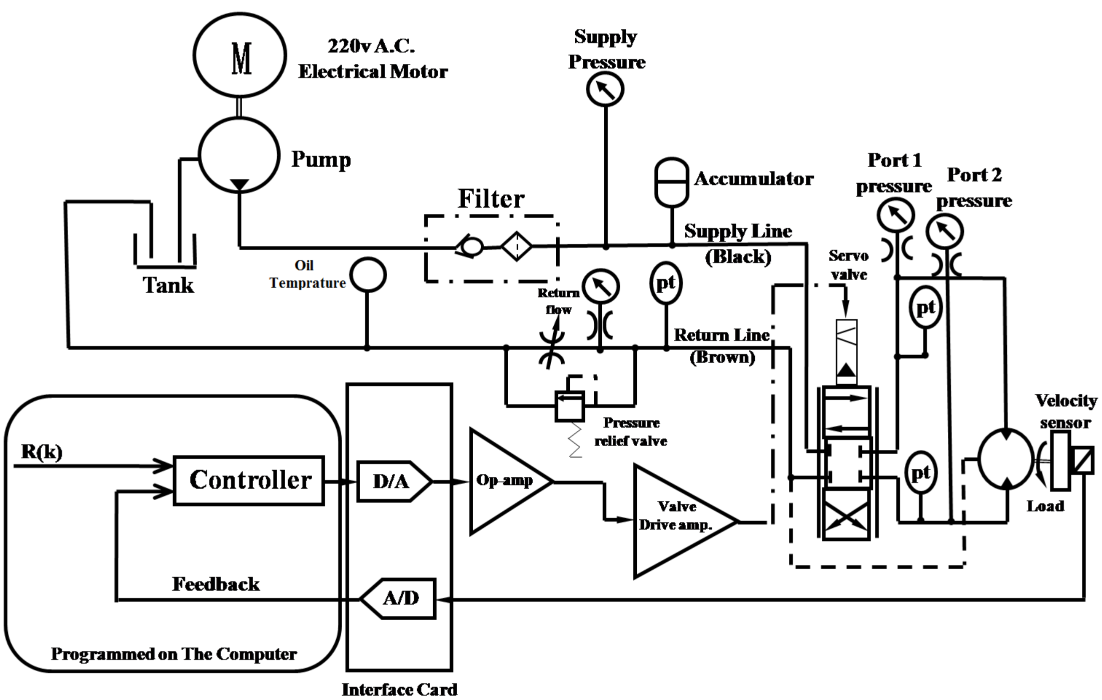


**Figure 2. Schematic diagram of the overall real time hydraulic control system.**

In this work, speed control is considered as the key characteristics of the desired behavior of the hydraulic actuator. In the electro-hydraulic system that is the control objective, to have an actuator's angular velocity follows a given reference input as closely as possible. This will be achieved through using adaptive GPC controller. The ARMA model parameters are obtained through the LS-SVM modeling algorithm. Using these parameters and employing the GPC algorithm to calculate the control signal. This control signal is fed to the electro-hydraulic servo-valve through the interface card. The accuracy of the GPC algorithm is mainly depending on the powerful of the modeling algorithm employed. As the modeling algorithm is powerful in capturing all the system phases and nonlinearities, the GPC controller make the actuator's angular velocity tracks the desired one. As LS-SVM is proven to be a powerful modeling algorithm to nonlinear systems so it is used here to accurately identify the system parameters.

## 3. LS-SVM Modelling

The nonlinear dynamic system with an input (*u*) and an output (*y*) can be described in discrete time by the NARX (nonlinear autoregressive with exogenous input) I/O model as [30]:

$$y(k+1) = f(x(k)) \tag{1}$$

where *f(.)* is the nonlinear function to be determined through LS-SVM, *y(k+1)* denotes the predicted output based on the identified model and *x(k)* is the repressor vector to be built once from the training set in off-line modelling. *x(k)* consisting of a finite number of past inputs and past outputs:

$$x(k) = \left[u(k-1), \cdots, u(k-n_u), \cdots, y(k), \cdots, y(k-n_y+1)\right]^T \tag{2}$$

The number of previous outputs (ny) and the number of previous inputs (nu) determine the dynamic order of the system.

In subsequent part the LS-SVM algorithm [9] will be briefly described for modelling nonlinear systems.

Using LS-SVM, it is desired to map the input data into a higher dimensional feature space via nonlinear mapping. In this higher dimensional feature space the decision function constructed is linear or nonlinear. The LS-SVM then maps the inner product of feature space to the original one via a kernel function.

Modelling based on LS-SVM regression based on a given set of N data points $\{x_i, y_i\}_{i=1}^N$ where the input of the system to be modeled is $x_i \in R^n$ and $y_i \in R^m$ is the corresponding output to be obtained via the LS-SVM model. The identified LS-SVM model is in the following form:

$$f(x) = w^T \cdot \phi(x) + b \tag{3}$$

where $\phi(x)$ is a nonlinear mapping from the input space *x* to a higher dimensional feature space. The SVM model parameters (*w* and *b*) are estimated based on empirical risk minimization (ERM) principle [5].

For LS-SVM [9] the optimization function can be obtained by using the squared loss function and equality constraints, which gives the following optimization function:

$$\begin{bmatrix} \min\limits_{w,b,e} J(w,e) = \frac{1}{2} w^T w + C \frac{1}{2} \sum_{k=1}^{N} e_k^2 \\ subject\ to \quad y_k = w^T \varphi(x_k) + b + e_k \ , \ k = 1, \ldots, N \end{bmatrix} \tag{4}$$

where $C$ is the regularization parameter to be selected, as it determines the trade-off between the empirical error and the regularized term and $e$ is the error between the actual and model outputs

This is equivalent to ridge regression [31] cost function formulated in the feature space. It must be noted that if $w$ have an infinite dimension the problem can't be solved. The Lagrangian is constructed in the dual space as:

$$L(w,b,e;\alpha) = J(w,e) - \sum_{k=1}^{N} \alpha_k \{ w^T \varphi(x_k) + b + e_k - y_k \} \tag{5}$$

where $\alpha_k$ are Lagrange multipliers to be estimated based on the $N$ training data set. To have the values $(\alpha_k, b)$, the conditions for optimality can be obtained by partial differentiating the Lagrangian with respect to each of the parameters obtaining:

$$\begin{bmatrix} \frac{\partial L}{\partial w} = 0 \ \rightarrow \ w = \sum_{k=1}^{N} \alpha_k \varphi(x_k) & \\ \frac{\partial L}{\partial b} = 0 \ \rightarrow \sum_{k=1}^{N} \alpha_k = 0 & \\ \frac{\partial L}{\partial e_k} = 0 \ \rightarrow \alpha_k = C e_{k,} & k = 1, \ldots, N \\ \frac{\partial L}{\partial \alpha_k} = 0 \rightarrow w^T \varphi(x_k) + b + e_k - y_{k,} & k = 1, \ldots, N \end{bmatrix} \tag{6}$$

It is clear from the above conditions; the values of the Lagrange multipliers are directly proportional to the error ($e_k$) at each sample ($k$). Eq.(9) have four parameters, eliminating $w$ and $e$, the LS-SVM is obtained as:

$$\begin{bmatrix} 0 & r_v^T \\ r_v & \Phi + I/C \end{bmatrix} \begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ y \end{bmatrix} \tag{7}$$

where $y=[y_1; \ldots; y_N]$, $r_v=[1, \ldots, 1_N]$, I is a $N \times N$ identity matrix, $\alpha=[\alpha_1; \ldots; \alpha_N]$ and $\Phi$ is a $N \times N$ symmetric matrix with the elements:

$$\Phi_{ij} = \varphi(x_i)\ \varphi(x_j) = K(x_i, x_j) \qquad i, j = 1, 2, \ldots, N \tag{8}$$

The LS-SVM model is then given by:

$$y(x) = \sum_{j=1}^{N} \alpha_j\ K(x, x_j) + b \tag{9}$$

where $\alpha_j \in R$ are Lagrange multipliers or shortly the main model parameters, $K(x, x_j)$ is any positive semi-definite kernel function satisfying Mercer's conditions [9, 32] and $b$ is the bias term . Eq.(10), indicates the solution of LS-SVM is given by a full rank square linear system that insures a unique solution which guarantee the convergence of that solution.

## 4. Sparse On-line LS-SVM

LS-SVM fits on-line training for its high speed, unlike neural networks and SVM, which indicates that it is suitable for on-line training as it achieves lower computational complexity, but there is a potential drawback for LS-SVM [6, 33]. The main drawback is that the sparseness of the data points is lost. Pruning LS-SVM is introduced here to solve this drawback. For on-line LS-SVM, the output vector *y(k)* and state matrix *x(k)* are variant which increase the computations time every sample, so the length of data set is set fixed. The sparseness is accomplished, the oldest data pair is omitted and in the same time a new data pair is added which is not used before with LS-SVM is equivalent to pruning LS-SVM introduced in [34]. Online training can be viewed as $\{x(k), y(k)\}$, $\Omega_{ij}(k)$, $\alpha(k)$ and $b(k)$ are functions of the sample ($k$). So the solution of LS-SVM can be rewritten as:

$$\begin{bmatrix} 0 & r_v^T \\ r_v & \Omega + I/C \end{bmatrix} \begin{bmatrix} b(k) \\ \alpha(k) \end{bmatrix} = \begin{bmatrix} 0 \\ y(k) \end{bmatrix} \tag{10}$$

and the LS-SVM model becomes:

$$y(x) = \sum_{j=1}^{N} \alpha_j(k)\ K(x, x_j(k)) + b(k) \tag{11}$$

which insures that the variations in the actual system dynamics are followed directly by LS-SVM model. The sparse online LS-SVM paradigm is utilized for modelling the hydraulic valve, hydraulic actuator and the tachogenerator as one unit at each sampling interval.

## 5. GPC Based on On-line LS-SVM

In this section GPC algorithm based on sparse on-line LS-SVM model using linear kernel function is introduced here briefly. The modeling parameter (regularization parameter) is identified through the simulation tests, which in turn utilized in practical experiments and the results in details will be explained down in the paper for the proposed controller as a speed one for the rotary hydraulic actuator. The model of the hydraulic system is obtained continuously from Eq.(11), which is a linear LS-SVM model which can be transformed into a linear I/O relation of the controlled system. The system output can be obtained at sample ($k$) as:

$$\begin{aligned} y(x) &= \sum_{j=1}^{N} \alpha_j(k)\{x_j^T(k)[u(k-1),\ldots,u(k-n_u), y(k-1),\ldots,y(k-n_y)]^T\} + b(k) \\ &= \sum_{j=1}^{N} \alpha_j(k)\{x_{j,1}(k)u(k-1) + \cdots + x_{j,n_u}(k)u(k-n_u) + x_{j,n_u+1}(k)y(k-1) + \cdots + x_{j,n_u+n_y}(k)y(k-n_y)]^T\} + b(k) \\ &= \sum_{j=1}^{N} [\alpha_j(k)(x_{j,1}(k) + \cdots + x_{j,n_u}(k)z^{-n_u+1})]u(k-1) + [\alpha_j(k)(x_{j,n_u+1}(k)z^{-1} + \cdots + x_{j,n_u+n_y}(k)z^{-n_y})]y(k) + b(k) \end{aligned} \tag{12}$$

which can be rewritten as a linear I/O equation as:

$$A(z^{-1})\, y(k) = B(z^{-1})u(k-1) + b(k) \tag{13}$$

where

$$A(z^{-1})=1-[\sum_{j=1}^{N}\alpha_j(k)(x_{j,n_u+1}(k)z^{-1}+\ldots+x_{j,n_u+n_y}(k)z^{-n_y})]$$

and $$B(z^{-1})=\sum_{j-1}^{N}\alpha_j(k)(x_{j,1}(k)+\ldots+x_{j,n_u}(k)z^{-n_u+1}) \quad (14)$$

which indicated that the two polynomials *A* and *B* depend on Lagrange multipliers produced from the LS-SVM model.

The GPC algorithm design is based on the following model [12, 23]:

$$A(z^{-1})\,y(k)=B(z^{-1})u(k-1)+v(k) \quad (15)$$

where $A(z^{-1})$ and $B(z^{-1})$ are polynomials in the backward shift operator $z^{-1}$. $v(k)$ can be decomposed as:

$$v(k)=v_{dc}+v_{ac}(k) \quad (16)$$

where, $v_{dc}$ is the direct-current component independent of time includes the bias term $b(k)$ in Eq.(11), while $v_{ac}$ is the alternating component whose mean is zero. Modeling $v_{ac}$ with $w(k)/\Delta$, Eq.(11) can be transformed into:

$$A(z^{-1})\,y(k)=B(z^{-1})u(k-1)+v_{dc}+w(k)/\Delta \quad (17)$$

where $w(k)$ is the disturbance or the noise with zero mean, $\Delta=1\text{-}z^{-1}$ is the difference operator, which introduced in the equation in order to provide an integral action and so, the steady-state error is eliminated. To obtain the $j$-step prediction of $y(k+j)$, the following Diophantine identity must be considered:

$$\begin{aligned}&1=E_j(z^{-1})A(z^{-1})\Delta+z^{-1}F_j(z^{-1}),\\&\deg E_j\ =\ j-1,\\&\deg F_j\ =n_a,G_j\ =E_jB\end{aligned} \quad (18)$$

where $E_j$ and $F_j$ are polynomials uniquely defined and the prediction interval $j$ is given. Multiplying Eq.(17) by $E_j\ \Delta z^j$ the following equation is obtained:

$$\begin{aligned}&E_j(z^{-1})A(z^{-1})\Delta\ y(k+j)\\&\quad=E_j(z^{-1})\,B(z^{-1})\,\Delta u(k+j-1)+E_j(z^{-1})\,(\Delta\,v_{dc}+w(t+j))\end{aligned} \quad (19)$$

as $v_{dc}$ is time independent, so $\Delta v_{dc}=0$. Then the above equation can be simplified as:

$$\begin{aligned}&E_j(z^{-1})A(z^{-1})\Delta\ y(k+j)\\&\quad=E_j(z^{-1})\,B(z^{-1})\Delta u(k+j-1)+E_j(z^{-1})\,w(k+j)\end{aligned} \quad (20)$$

substituting for $Ej(z^{-1})A(z^{-1})\ \Delta$ we obtain:

$$y(k+j) = E_j(z^{-1})B(z^{-1})u(k+j-1) + F_j(z^{-1})y(k) + E_j(z^{-1})\,w(k+j) \quad (21)$$

then the multistep ahead predictor of $y(k)$ can be given as:

$$\hat{y}(k+j\,|\,k) = G_j(z^{-1})\Delta u(k+j-1) + F_j(z^{-1})y(k) \quad (22)$$

where $\hat{y}(k+j\,|\,k)$ is the optimal predictor in the prediction interval $j$,

$$G_j(z^{-1}) = E_j(z^{-1})B(z^{-1}) = \frac{B(z^{-1})[1-z^{-j}F_j(z^{-1})]}{A(z^{-1})\Delta} \quad (23)$$

The GPC control law can be obtained through minimizing the following cost function:

$$J(N_1,N_2,N_u) = \sum_{l=N_1}^{N_2}[\hat{y}(k+l) - y_r(k+l)]^2 + \sum_{l=1}^{N_u}\lambda(l)[\Delta u(k+l-1)]^2 \quad (24)$$

where $y_r(k)$ is the set-point, $N_1$,$N_2$ and $N_u$ are the minimum, maximum prediction horizons and control horizon respectively (e.g. $\Delta u(k+l) = 0, l \geq N_u$), and $\lambda(l)$ is the control-weighting sequence, where the choice of these parameters is in details in [12,35]. Then the $j$-step ahead predictor $y(k+j)$ can be expressed in the vector form as:

$$y(k+j) = Gu(k) + Fy(k) + H\Delta u(k-1) \quad (25)$$

and the corresponding control law is given by:

$$\Delta u(k) = [1,0,0,\cdots,0](G^TG+\lambda I)^{-1}G^T[y_r - Fy(k) + H\Delta u(k-1)] \quad (26)$$

where the variables in Eqs.(25 and 26) are clearly defined by Clarke [12,13]. Eq.(26) can be rewritten as:

$$\Delta u(k) = d^T(y_r - y_0) \quad (27)$$

Where

$$y_r = [y_r(k+1),\cdots,y_r(k+N_2)]^T,$$
$$y_0 = [y_0(k+1\,|\,k),\cdots,y_0(k+N_2\,|\,k)],$$
$$d^T = [1,0,0,\cdots,0](G^TG+\lambda I)^{-1}G^T,$$

and

$$G = \begin{bmatrix} g_0 & & \cdots & 0 \\ g_1 & & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ \vdots & \cdots & \cdots & g_0 \\ g_{N_2-1} & g_{N_2-2} & \cdots & g_{N_2-N_u} \end{bmatrix}$$

Here, $y_o(k+j|k)$ is the free response when the future control increments are all zeros, $g_i$ is the step response coefficient of the model.

## 6. Stability Properties of Open Loop and Closed Loop GPC

In [36,37], the stability properties of the closed-loop system using the GPC controller are studied through the following theorems:

*Theorem V.1.* If $\lambda > 0$, there must exist a unique optimal control law for GPC. This result can be easily obtained, so non-zero control weighting can always guarantee a unique optimal control law.

*Theorem V.2.* For the n[th] order plant, if $N_u<n+1$, $N_2> N_1+N_u-1$, there exists a unique optimal control Law for GPC. It can be derived from the property of Markov parameters directly.

*Theorem V.3.* For the n[th] order plant, if $N_2 \geq n+2, N_1 \geq n+1$ there exists no unique optimal control law of GPC.

The above theorems discuss the solvability issue in all the cases of $N_1$ and $N_u$ completely. It should be noted that, $N_2$ can be selected as $N_2 \geq N_1 + N_u - 1$ just to satisfy the necessary condition of $G$ to be full rank in column. All these discussions could help to choose the proper tuning parameters and guarantee the further discussion meaningful.

As proposed in [38], the robustness properties of the closed-loop system using the GPC controller are studied as:

Let us define these new symbols as:

$$A_k\left(z^{-1}\right)=1+a_1^* z^{-1}+\cdots+a_{n_a+1}^* z^{-\left(n_a+1\right)},$$
$$B_k\left(z^{-1}\right)=d^T\left[F_1\left(z^{-1}\right),\cdots,F_{N_2}\left(z^{-1}\right)\right]^T = f_0 + f_1 z^{-1} + \cdots + f_{n_a} z^{-n_a} \tag{28}$$

The coefficients of $A_c(z^{-1})$ are determined by:

$$\begin{bmatrix} 1 \\ a_1^* \\ \\ \vdots \\ \\ a_{n_a}^* \end{bmatrix} = \begin{bmatrix} 1 & & \cdots & & 0 \\ c_2 - 1 & 1 & \ddots & & 0 \\ c_3 - c_2 & c_2 - 1 & & \ddots & \vdots \\ \vdots & \cdots & \cdots & & 1 \\ c_{n_a+2} - c_{n_a+1} & \cdots & \cdots & & c_2 - 1 \end{bmatrix} \begin{bmatrix} 1 \\ a_1 \\ \\ \vdots \\ \\ a_{n_a} \end{bmatrix}, \tag{29}$$

Where

$$c_i = \sum_{j=1}^{N_2} d_j g_{i+j-1} \tag{30}$$

*Theorem V.4.* $\mathbf{A}_k - \mathbf{z}^{-1}\mathbf{B}_k\mathbf{B}$ can be divided exactly by $\Delta A$, where $\Delta A = \left(1 - z^{-1}\right)A$.

*Proof.* Suppose $q$ is a root $\Delta A$ of repeated $\beta(\beta \geq 1)$ times and this root is repeated as:

$$\Delta A\left(q^{-1}\right)=0,\cdots,\Delta A^{[\beta-1]}\left(q^{-1}\right)=0 \tag{31}$$

where $\Delta A^{[\beta-1]}$ is the $(\beta-1)^{th}$ derivative of $\Delta A\left(z^{-1}\right)$ where all the derivatives are with respect to $z^{-1}$.

This theorem can be proved through proving through verifying that:

$$\Delta A\left(q^{-1}\right)-q^{-1}B_k\left(q^{-1}\right)B\left(q^{-1}\right)=0 \tag{32}$$

and the $\ell_{th}$ derivative of $\mathbf{A_k - z^{-1}B_kB}$ equals zero at the point $z=q$ where $\ell \le \beta - 1$.

From Eqs. (28 and 32), the following equation is obtained:

$$A_k\left(q^{-1}\right)=c_2q^{-1}+\cdots+[\left(c_{n_a+2}-c_{n_a+1}\right)+\cdots+c_2a_{n_a}]\,q^{-(n_a+1)}=0 \tag{33}$$

From Eq.(15), this equality is obtained:

$$F_j\left(q^{-1}\right)=q^j \tag{34}$$

From Eqs.(26 and 33), $B_F\left(q^{-1}\right)$ can be obtained as:

$$B_k\left(q^{-1}\right)=\sum_{j=1}^{N_2}d_jq^j \tag{35}$$

The coefficients of $A(z^{-1})$, $B(z^{-1})$ and $G(z^{-1})$ must meet the following characteristics of step response coefficients, which will be used in the rest of the proof:

$$\begin{cases} g_0=b_0 \\ \left(g_1-g_0\right)+g_0a_1=b_1 \\ \vdots \\ \left(g_{n_b-1}-g_{n_b-2}\right)+\left(g_{n_b-2}-g_{n_b-3}\right)a_1+\cdots+\left(g_{n_b-n_a-1}-g_{n_b-n_a-2}\right)a_{n_a}=b_{n_b} \\ \left(g_{i-1}-g_{i-2}\right)+\left(g_{i-2}-g_{i-3}\right)a_1+\cdots+\left(g_{i-n_a-1}-g_{i-n_a-2}\right)a_{n_a}=0,\ i\ge n_b \end{cases} \tag{36}$$

where $g_i=0$ for $i<0$.

Now we will prove Eq.(30) through mathematical induction as:

*When $N_2=1$, the equation to be proved can be rewritten as:*

$$\begin{aligned} &d_1[g_1q^{-1}+\left[\left(g_2-g_1\right)+g_1a_1\right]q^{-2}+\cdots+ \\ &\left[\left(g_{n_a+1}-g_{n_a}\right)+\cdots+g_1a_{n_a}\right]q^{-(n_a+1)}]=d_1B\left(q^{-1}\right) \\ &\qquad =d_1\left(b_0+b_1q^{-1}+\cdots+b_{n_b}q^{-n_b+1}\right) \end{aligned} \tag{37}$$

substituting for $b_i$ from Eq.(34), then the above equation transformed to:

$$d_1g_0\Delta A\left(q^{-1}\right)=0 \tag{38}$$

From Eq.(31), the above equation holds.

*Assume that $N_2=t$. For $N_2=t+1$, let $\delta A$ and $\delta B_F$ are the increments of $A_c$ and $B_F$ produced by $d_{t+1}$ respectively, and then it is needed to prove that $\delta A_k\left(q^{-1}\right)-q^{-1}\delta B_k\left(q^{-1}\right)B\left(q^{-1}\right)=0$ which can be rewritten as:*

$$\begin{aligned} &d_{t+1}\,g_{t+1}\,q^{-1}+\left[\left(g_{t+2}-g_{t+1}\right)+g_{t+1}a_1\right]q^{-2}+\cdots \\ &+\left[\left(g_{t+n_a+1}-g_{t+n_a}\right)+\cdots+g_{t+1}\,a_{n_a}\right]q^{-(n_a+1)}]=d_{t+1}\,B\left(q^{-1}\right) \\ &\qquad =d_1\left(b_0+b_1\,q^{-1}+\cdots+b_{n_b}q^{-n_b+1}\right) \end{aligned} \tag{39}$$

Which can be transformed as in Eq.(36) to be:

$$d_1\left(g_0 + g_1 q^{-1} + \cdots + g_q q^{-q}\right)\Delta A\left(q^{-1}\right) = 0, \tag{40}$$

Which can be obtained directly from Eq.(31). From Eqs.(38 and 39), the condition of Eq.(32) holds. If $\beta > 1$, it is also needed to prove that the $(\beta - 1)^{th}$ derivatives of $\mathbf{A_k} - \mathbf{z^{-1}B_k B}$ are all zeros at the point $z=q$.

*Proof*. The Diophantine Identity defined in Eq.(18) can be rewritten as:

$$z_j = z_j E_j \Delta A + F_j, \tag{41}$$

By computing the $\ell_{\text{th}}$ derivative on both sides of Eq.(39) the following equation is obtained for $\ell \le \beta - 1$:

$$F_j^{[\ell]}\left(q^{-1}\right) = (-1)^{\ell} * j * (j+1) \cdots (j + \ell + 1) * q^{j+\ell+1} \tag{42}$$

Combining this equation with Eq.(31) and from Eq.(34), the $\ell_{\text{th}}$ derivative of $F_j\left(z^{-1}\right)$ at $z=q$ is equal to the derivative of $z^j$. If $R\left(z^{-1}\right) = \sum_{i=1}^{N_2} d_i z_i$, using this the following relation is obtained:

$$B_k\left(q^{-1}\right) = R\left(q^{-1}\right), \cdots, B_k^{[\beta-1]}\left(q^{-1}\right) = R^{[\beta-1]}\left(q^{-1}\right) \tag{43}$$

Computing the the $\ell_{\text{th}}$ derivative of $\mathrm{A_k - z^{-1}B_k B}$. From Eq.(43), it is clear that both $\mathrm{A_k - z^{-1}B_k B}$ and $\mathrm{A_k - z^{-1}RB}$ have the same derivative values at the point $z=q$. From Eqs.(42 and 43), it is clear that the polynomial contains $\Delta A$, so the corresponding $\ell_{th}$ derivatives are zeros which completes the proof of theorem V.4.

The result from Theorem V.4 ensures directly that:

*Corollary V.1*. If $n_b=l$, then $\mathrm{A_k - z^{-1}B_k B} = \left(1 - \mathrm{z^{-1}}\right)A$.

The closed-loop transfer function using the GPC control law can be deduced as:

$$\frac{\mathrm{y}\left(\mathrm{z^{-1}}\right)}{\mathrm{y_r}\left(\mathrm{z^{-1}}\right)} = \frac{\mathrm{d_s z^{-1}} A \mathrm{B_P}}{A_P\left(A_k - \mathrm{z^{-1}B_k B}\right) + \mathrm{z^{-1}B_k AB}} \tag{44}$$

where the numerator and denominator of the real plat are described as:

$$\begin{aligned}
A_P\left(z^{-1}\right) &= 1 + a_1^{\bullet} z^{-1} + \cdots + a_{nx}^{\bullet} z^{-nx} \triangleq A + \delta A, \\
B_P\left(z^{-1}\right) &= b_0^{\bullet} + b_1^{\bullet} z^{-1} + \cdots + b_{ny}^{\bullet} z^{-ny+1} \triangleq B + \delta B, \\
\delta A\left(z^{-1}\right) &= \delta a_1 z^{-1} + \cdots + \delta a_{nx} z^{-nx}, \ and \\
\delta B\left(z^{-1}\right) &= \delta b_0 z^{-1} + \delta b_1 z^{-2} + \cdots + \delta b_{ny} z^{-ny+1},
\end{aligned} \tag{45}$$

where $n_x$ and $n_y$ are already known. The characteristic polynomial $D_m$ is obtained as:

$$\begin{aligned}
D_m &= A_P A_k + \mathrm{z^{-1}B_k}\left(\mathrm{AB_P} - \mathrm{A_P B}\right) \\
&= A_k A + \left[A_k \delta A + \mathrm{z^{-1}B_k}\left(\mathrm{A} \cdot \delta \mathrm{B} - \delta \mathrm{A} \cdot \mathrm{B}\right)\right]
\end{aligned} \tag{46}$$

The condition related to stability of the real closed-loop system, namely the problem of robustness will be presented in the following theorem. Firstly we assume that the model's numerator does not contain zeros on the unit circle.

*Theorem V.5.* Consider the GPC designed by some methods, which leads to a stable controller $A_c(z^{-1})$. Then the real closed-loop remains stable if:

$$\left| A_k\left(\frac{1}{A}-\frac{1}{A_P}\right)+z^{-1}B_F\left(\frac{B_k}{A_P}-\frac{B}{A}\right)\right|_{z=e^{jw}} < \left|\frac{A_k}{A_P}\right|_{z=e^{jw}} \tag{47}$$

For any real $w\in[0,\pi]$.

*Proof*. If the condition of Eq.(47) is satisfied, then from Eq.(46), it is clear that the characteristic polynomial $D_m$ and $A_k A$ have the same number of roots outside the unit circle Rouche's theorem.

*Theorem V.6.* (Rouche's Theorem) Let the bounded region *D* have as its boundary a simple closed contour C [39]. Let *f(z)* and *g(z)* be analytic both in *D* and on C. Assume that $|f(z)| < |g(z)|$ on *C*. Then *f(z) - g(z)* has in *D* the same number of zeros as *g(z)*, all zeros counted according to their multiplicity.

On the other hand, $\mathrm{A_k - z^{-1}B_kB}$ can be divided exactly by *A* from Theorem 5.4, hence there is no harm to write $\mathrm{A_k - z^{-1}B_kB = A\cdot V}$. From preconditions, only *A* probably contains unstable poles, but they can be fully reduced by the numerator, so the real closed-loop is stable. The uncertain part in the square brackets of Eq.(46), its upper bound on the unit circle can be estimated as follows:

$$\left|A_k\delta A+\mathrm{z^{-1}B_k}\left(\mathrm{A}\cdot\delta\mathrm{B}-\delta\mathrm{A}\cdot\mathrm{B}\right)\right|_{z=e^{jw}} \le \left\{\left(S^2+T^2\right)\cdot n\cdot\left(\sum_{i=1}^{n_x}\delta a_i^2+\sum_{i=1}^{n_y}\delta b_i^2\right)\right\}^{1/2} \tag{48}$$

where

$$S=1+\sum_{i=1}^{n_a+1}\left|a_i^*\right|+\sum_{i=0}^{n_a}|f_i|\cdot\sum_{i=0}^{n_b}|b_i|,$$
$$T=\sum_{i=0}^{n_a}|f_i|\left(1+\sum_{i=1}^{n_a}|a_i|\right),\ n=\max\left(n_x,n_y\right) \tag{49}$$

If $A_k$ is stable, continuous function $A_k(e^{-jw})A(e^{-jw})$ must have a non-zero minimum module on the unit circle, denoted as $\inf_{w\in[0,2\pi]}\left|\mathrm{A_k}\left(\mathrm{e^{-jw}}\right)\mathrm{A}\left(\mathrm{e^{-jw}}\right)\right|$.

The uncertain radius $\delta R$ can be obtained as:

$$\delta R=\left(\sum_{i=1}^{n_x}\delta a_i^2+\sum_{i=1}^{n_y}\delta b_i^2\right)^{1/2} \tag{50}$$

From these results corollary V.1 can be inferred.

*Theorem V.7.* If the conditions of Theorem V.5 are satisfied and

$$\delta R<\inf_{w\in[0,2\pi]}\left|A_k\left(e^{-jw}\right)A\left(e^{-jw}\right)\right|/\left[n\left(S^2+T^2\right)\right]^{1/2} \tag{51}$$

then the closed loop system is stable.

## 7. Practical Results

The LS-SVM paradigm is employed to obtain a model based on practical data utilizing polynomial kernel function. The results obtained in simulation trials shows that the best regularization parameter is C=*30* and the control weighting parameter is $\lambda=10$ where the other controller parameters are $N_1=1$, $N_2=8$, $N_u=8$ following the rules indicated in [12,35] for selecting these parameters. In this case the sparseness in obtained through fixing the regression matrix to a definite size which here is selected to be 10 taking two inputs $(n_u=2)$ and two previous outputs $(n_y=2)$ in the regression vector. The GPC controller with parameters as the simulation ones is implemented based on sparse online LS-SVM using linear kernel function. The obtained sparse LS-SVM model is identified continuously are each sample which is then employed to have the polynomials $A(z^{-1})$ and $B(z^{-1})$ with the order indicated above. All the results are obtained through Basic programming which speeds the calculations.

The hydraulic actuator's speed response for step changes is shown in Fig.3(a) and its corresponding control signal is depicted in Fig.3(b). From this result, the proposed controller based on sparse LS-SVM can be employed practically for speed control of hydraulic actuator. Comparing to self-tuning PID controller, it can be conducted that the later one has a great overshoot at step changes due to the great time required for it to adapt its model parameters.

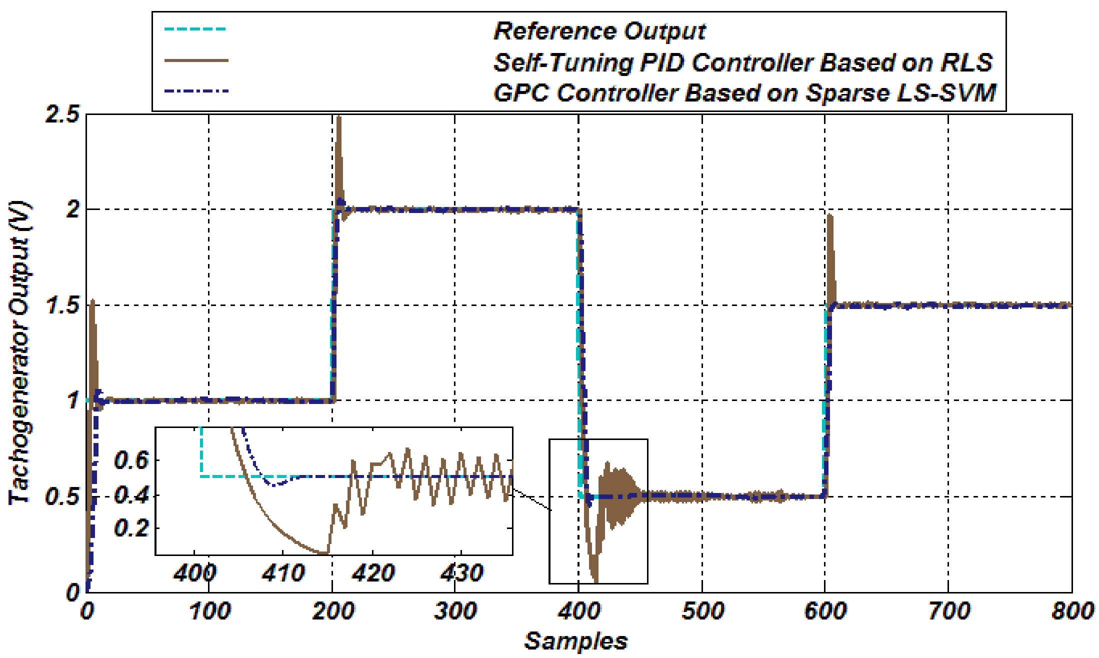


**Figure 3(a). The GPC results versus the self-tuning PID for variable set changes.**

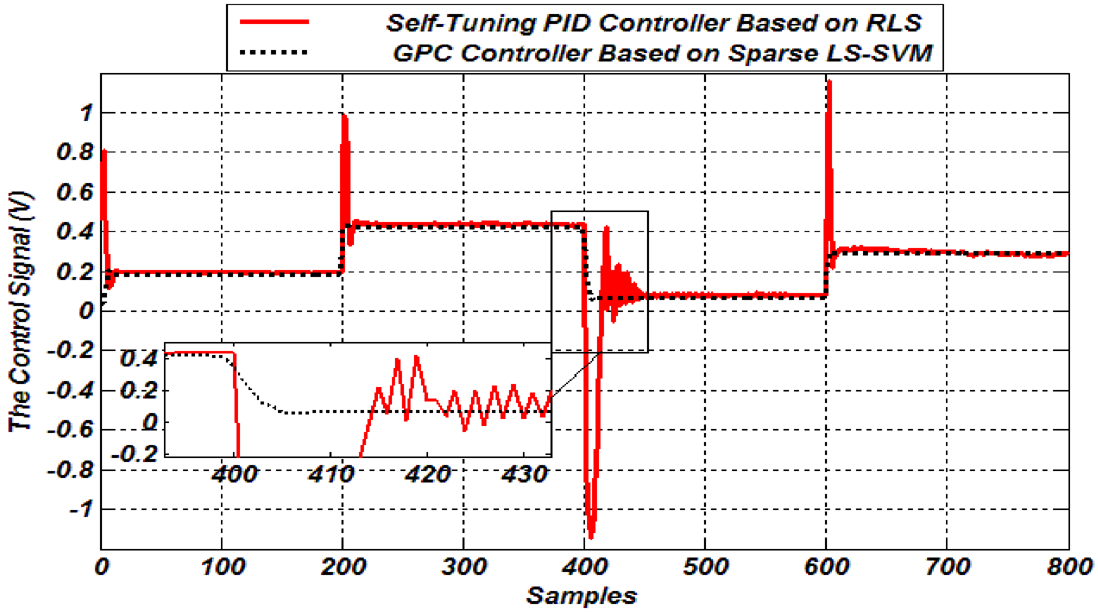


**Figure 3(b). The control signals for both the controllers.**

Fig.4(a), indicates the tachogenerator output versus the desired one under 20% fixed load disturbance where Fig.4(b) indicates the corresponding control signal. The proposed controller possesses a good response for fixed load disturbance. From Fig.4(b) shows that the control signal in the case of GPC is less than that of self-tuning PID which shows the robustness of the proposed controller to load disturbances. Also, the PID controller can't bear this load which is clearly in the figure as it tries to adapt itself through overshooting the output but the control signal still increases.

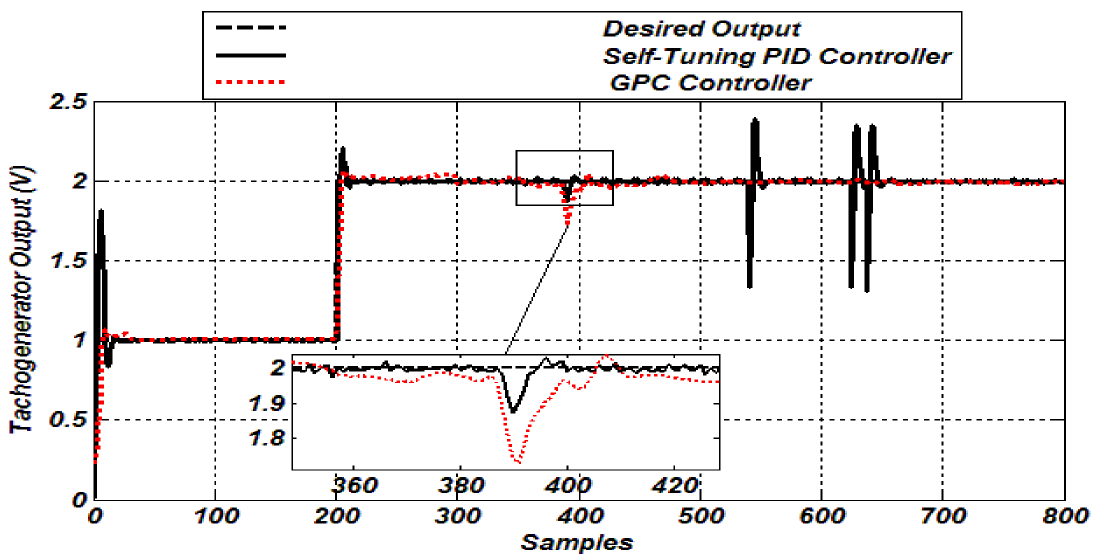


**Figure 4(a). The practical results of both the controller for fixed load disturbance.**

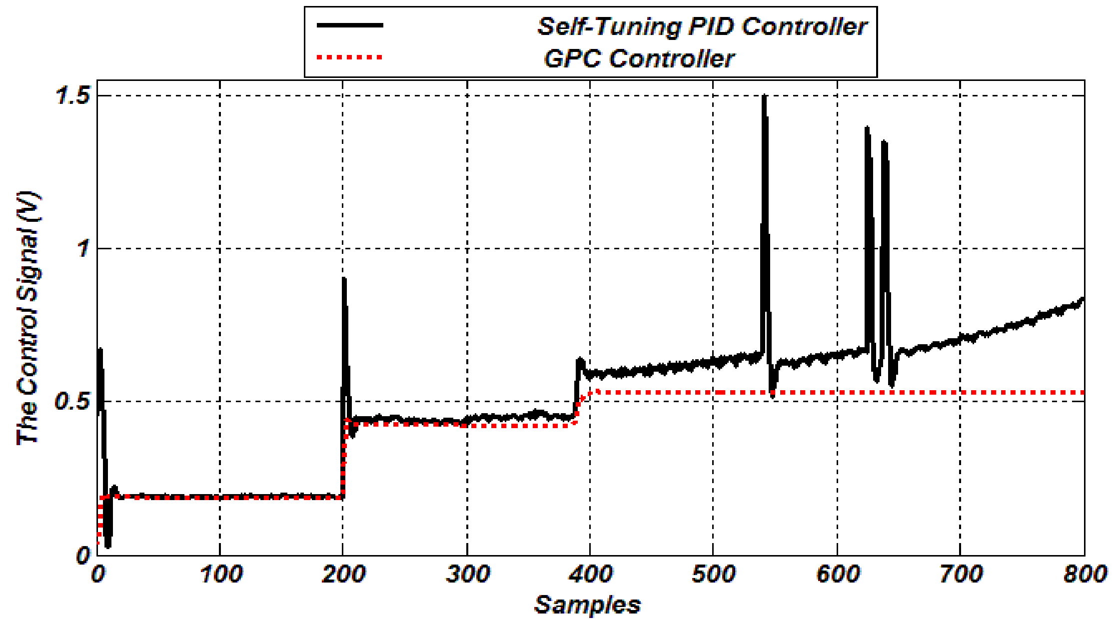


**Figure 4(b). The control signals of the two controllers.**

As for fixed load disturbance the hydraulic actuator is experienced to variable disturbance as depicted in Fig.5(a) where its corresponding control signal is depicted in Fig.5(b). The proposed controller has a good response under large load disturbance. From the results of variable load it can noticed that the PID controller takes more time than GPC to modify its output to the desired one in both cases when applying the load and removing it. This is due to the time taken by the PID to modify its parameters is long than that taken by the proposed controller.

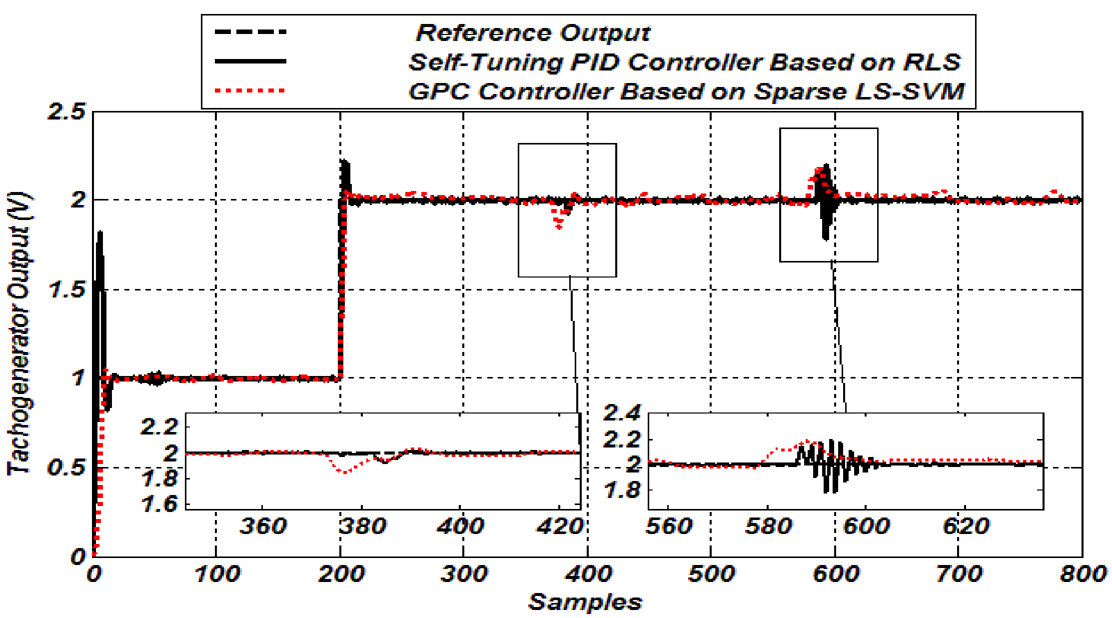


**Figure 5(a). The results of the two controllers under variable load disturbance.**

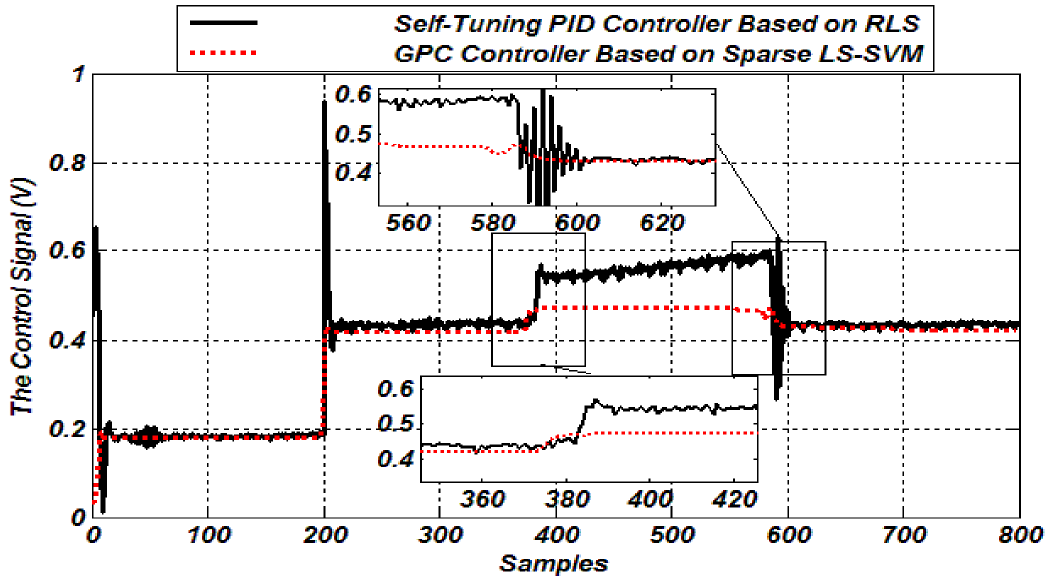


**Figure 5(b). The corresponding control signals when applying variable load.**

From these results, it is clear that the proposed controller based on sparse LS-SVM model is a good speed controller for the rotary hydraulic actuator. Model reference control algorithm based on sparse LS-SVM presented in this contribution is feasible in real-time control systems especially their response is very fast like the one under study which insures the applicability of this algorithm for other systems with slower response; also it shows the performance of the LS-SVM paradigm and finally the feasibility of the sparseness method used. It is significant that the GPC controller based on sparse LS-SVM model can be used effectively as a speed controller for the rotary hydraulic actuator under variable operating conditions and load disturbances.

The performance of the hydraulic motor under different operating conditions (variable set-points, fixed and variable loads) using an intelligent self-tuning PID-controller based on RLS algorithm and GPC controller based on sparse LS-SVM are evaluated on the basis of settling time, maximum overshoot, steady state error, immunity to load disturbances and maximum load disturbance as indicated in Table 1. From the results indicated in Table 1, it is clear that the self-tuning PID controller is fast as the proposed controller but with a great overshoot, also the proposed controller can bear larger loads and is less sensitive to these load disturbances which means the smaller load cause a change in the shaft output speed.

**Table 1. Comparison Between Speed Performances of the Proposed Controller Based on Sparse LS-SVM and Self-tuning PID Based on RLS.**

| **Comparison** | **Self-Tuning PID-Controller Based on RLS Algorithm** | **Adaptive GPC Controller Based on LS-SVM** |
|---|---|---|
| Settling time (ts) sec. | 0.1 | 0.1 |
| Maximum overshoot (mp) | 1.05 | 0.1 |
| Steady state error (ess) | 0.0025 | 0.0015 |
| immunity to Load Disturbance | Very Sensitive to load | less Sensitive to load |
| Maximum Load Disturbance | 25% | 65% |

## 8. Conclusion

The main contribution of this paper is to show the feasibility of the GPC controller based on the new sparseness method for LS-SVM for speed control of a rotary hydraulic actuator. This can be achieved through using the LS-SVM paradigm to model the hydraulic valve, actuator and tachogenerator as one unit continuously at each sample. The LS-SVM parameters are then used to obtain the ARMA model polynomials. The sparseness of LS-SVM is reached through adding new information and deletes the oldest one. The presented GPC algorithm can construct a controller with high precision and satisfy the demand of real-time property. The proposed controller is implemented practically with parameters as the simulated ones to control the rotary hydraulic actuator's speed. The results of the presented controller show its effectiveness and feasibility. The linear kernel function is employed here for modelling using LS-SVM instead of RBF in order to avoid the nonlinear programming problem to be resolved in each sampling interval. The performance of the presented controller is compared to that of self-tuning PID controller with model polynomials obtained based on the RLS paradigm. The comparison shows the feasibility of the presented controller for the speed control of the rotary hydraulic actuator under different operating conditions and load disturbances.

## References

[1] C.M. Bishop, “Neural Networks for Pattern Recognition,” *Oxford University Press*, 1995.

[2] J. Valyon, “Selection Methods for Extended Least Squares Support Vector Machines,” *International J. of Intelligent Computing and Cybernetics*, 1756-378X, pp. 69-93, 2008

[3] K. Hornik, M. Stinchcombe and H. White, “Multilayer Feedforward Networks are Universal Approximators,” *Neural Networks J.*, vol. 2, pp. 359–366, 1989.

[4] V.N.Vapnik, *The Nature of Statistical Learning Theory.* New York: Springer-Verlag, 1995.

[5] V.N.Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1998.

[6] J.A.K. Suykens J. De Brabanter, L. Lukas and J. Vandewalle, “Weighted Least Squares Support Vector Machines: Robustness and Sparse Approximation,” *Neurocomputing J.*, vol. 48, pp. 85–105, 2002.

[7] C. Saunders, A. Gammerman and V. Vovk, “Ridge Regression Learning Algorithm in Dual Variables,” *Proceedings of the 15th International Conference on Machine Learning, (ICML’98)*, pp. 515–521, 1998.

[8] J.A.K. Suykens and J. Vandewalle, “Least Squares Support Vector Machine Classifiers,” *Neural Process. Letter*, vol. 9, No. 3, pp. 293–300, 1999.

[9] J. A. K. Suykens, T. V. Gestel, J. D. Brabanter, B. D. Moor and J. Vendewalle, *Least Squares Support Vector Machines*. Singapore, World Scientific, 2002.

[10] J. Valyon and G. Harvath, “A Robust LS-SVM Regression,” *Proceeding of World Academy of Science, Engineering and Technology*, vol. 7, pp. 148-153, 2005.

[11] L. Hoegaerts, J. A. K. Suykens, J. Vandewalle and B. D. Moor, “A comparison of pruning algorithms for sparse least squares support vector machines,” *Lecture Notes in Computer Science*; vol. 3316, pp. 1247-1253, 2004.

[12] D.W. Clarke, C. Mohtadit and P. S. Tuffs, ”Generalized Predictive Control Part I. The Basic Algorithm,” *Automatica*, vol. 23, No. 2, pp. 137-148, 1987.

[13] D.W. Clarke, C. Mohtadit and P. S. Tuffs, ”Generalized Predictive Control Part II. Extensions and Interpretations,” *Automatica,* vol. 23, No. 2, pp. 149-160, 1987.

[14] F. Allgower and A. Zheng, *Nonlinear Model Predictive Control* Germany, Birkhauser Verlag, 2000.

[15] D.W. Clarke, “Application of Generalized Predictive Control to Industrial Processes,” *IEEE Control Systems Magazine,* vol. 8, pp. 49-55, 1988.

[16] S. Huang, K.K. Tan, T.H. Lee, *Applied Predictive Control*, London, Springer-Verlag, 2002.

[17] B. Kouvaritakis and M. Cannon, *Nonlinear Predictive Control-Theory and Practice*, London, IEE, 2001.

[18] C. Lu and C. Tsai, "GPC Using Recurrent Fuzzy Neural Networks for Industrial Processes," *Journal of Process Control,* vol. 17, pp. 83–92, 2007.

[19] J.B. Rawlings, "Tutorial Overview of Model Predictive Control," *IEEE Control Systems Magazine*, vol. 4, pp. 38–52, 2000.

[20] D. Darryl and G. Martin, "A New Real-Time Perspective on Nonlinear Model Predictive Control," *Journal of Process Control,* vol. 16, No. 6, pp. 635-643, 2006.

[21] S.J. Qin and T.A. Badgwell, "A Survey of Industrial Model Predictive Control Technology," *Control Engineering Practice,* vol. 11, No. 7, pp. 733-764, 2003.

[22] L. Juan, S. Hong-Ye and C. Jian, "Generalized Predictive Control with Online Least Squares Support Vector Machines," *Acta Automatica Sinica,* vol. 33, No. 11, pp. 1182-1188, 2007.

[23] L. Bin, S. Hong-ye and C. Jian, "New Predictive Control Algorithms Based on Least Squares Support Vector Machines," *Journal of Zhejiang University Science,* pp. 440-446, 2005.

[24] B. Liu, H. Su, W. Huang and J. Chu, "Temperature Prediction Control Based on Least Squares Support Vector Machines," *Journal of Control Theory and Applications,* vol. 4, pp. 365-370, 2004.

[25] D. Yongbin, Y. Weidong, W. Shaofu , Z. Ming and L. Qinghua, "Generic Model Predictive Control Strategy Based on Integrated Weighted Least Square Support Vector Machines," *Proc. of the 2nd International Symposium on Computational Intelligence and Design,* vol. 2, pp. 216-220, 2009.

[26] S. Iplikci, "Support vector machines-based generalized predictive control," *Int. J. Robust & Nonlinear Control,* vol. 16, pp. 843–862 2006.

[27] R. Poley, "DSP Control of Electro-Hydraulic Servo Actuators," Application Report, *Texas Instruments*, SPRAA76, 2005.

[28] H. Yousefi , H. Handroos and A. Soleymani, "Application of Differential Evolution in System Identification of a Servo-Hydraulic System with a Flexible Load," *Mechatronics*, vol. 18, pp. 513-528, 2008.

[29] H. Forsterling and P. Furtner, "Application of Hydraulic Actuators for Damping and Isolation of Structural Oscillations," *Proceeding of the Third European Conf. on Structural Control*, 3ECSC, 2004.

[30] X. Wang, W.Liang, X. Cai, G. Lv, C. Zhang and H. Zhang, "Application of Adaptive Least Square Support Vector Machines in Nonlinear System Identification," *Proc. of the 6th World Congress on Intelligent Control and Automation,* vol. 1, pp. 1897-1900, 2006.

[31] G. H. Golub and C. F. Van Loan, *Matrix Computations, Baltimore MD: Johns Hopkins University Press.* 1989.

[32] C. J. C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition" *Data Mining and Knowledge Discovery*, vol. 2, No. 2, pp. 121–167, 1998.

[33] L. Wei, Z. Chen, J. Li and W. Xu, "Sparse and Robust Least Squares Support Vector Machine: a Linear Programming Formulation," *Proc. of IEEE Int. Conf. on Grey Systems and Intelligent Services,* pp. 1134-1138, 2007.

[34] J.A.K. Suykens, L. Lukas, and J. Vandewalle, "Sparseness Approximation Using Least Squares Support Vector Machines," *IEEE int. symposium on Circuits and System, vol. 2, pp. 757-760,* 2000.

[35] D. W. Clarke and C. Mohtadi, "Prosperities of Generalized Predictive Control," *Automatica,* vol. 25, No. 6, pp. 859-875, 1989.

[36] X. Yugeng and J. Zhang, "Study on The Closed-loop Properties of GPC," *Science in China (Series E),* vol. 40, No. 1, pp. 54-63, 1997.

[37] J. A. Rossiter, B. Kouvaritakiss and J. R. Gossners, "Feasibility and Stability Results for Constrained Stable Generalized Predictive Control," *Automatica*, vol. 31, No. 6, pp. 86-877, 1995.

[38] S. Mingwei C. Zengqiang and Y. Zhuzhi, "The Principle of Robustness in Generalized Predictive Control," *Appl. Math. J. Chinese Univ. Ser. B,* vol. 14, No. 3, pp. 293-300, 1999.

[39] I. J. B. F. Adan, J. S. H. Leeuwaarden, and E. M. M. Winands, "On the Application of Rouche's Theorem in Queuing Theory," *Operations Research Letters*, vol. 34, pp. 355-360, 2006.

# Authors

**Hossam Mohammad Khalil**, received his B.Sc. in Industrial Electronics and Control Engineering from Faculty of Electronic Engineering, Minoufiya University on 2004. He was in the 1st place on his department with excellent grade. He gained M.Sc. degree in Electronic Engineering, Automatic Control Engineering and Measurements department from Faculty of Electronic Engineering, Minoufiya University on 2010. Now he is an assistant lecturer and PhD Student in the same faculty.