Predictive Control Realization in Delta Domain for Automotive Powertrain System and Finite Word Length Representation

C. Budaciu, C. Lazar

Department of Automatic Control and Applied Informatics
Technical University "Gheorghe Asachi" of Iasi
Blvd. Mangeron, 27, 700050, Iasi, Romania
{chalauca, clazar}@ac.tuiasi.ro

Abstract: The automotive systems contain fast processes that imply fast sampling rate. Moreover, the existing digital hardware contains processing registers that have finite precision and therefore, the representation can be restricted to a finite number of bits. Thus, the controller must be implemented on these limited word length control units in the context of fast sampling rate. Unfortunately, the classical discrete time representation may become unstable due to finite word lengths calculations. A new solution proposed in this paper is to design a predictive control strategy in the discrete δ-domain. State Space δ Generalized Predictive Control, in particular, referred as SS δ GPC algorithm, can bring significant improvements, especially when is implemented on embedded systems. In this paper, the proposed solution of predictive controller designed in δ-domain was successfully applied to automotive powertrain system and demonstrates a large potential in finite word length implementation.

Keywords: fast sampling, finite word length, δ - operator, predictive control.

1. INTRODUCTION

Nowadays, due to the performance requirements and efficiency, the automotive powertrain control systems must include control strategies implemented using finite precision hardware. The control law implementation using fixed point arithmetic provides lower memory requirements, power consumption and higher speed in comparison to floating point representation. Furthermore, the automatic control strategy plays an important role in automotive powertrain systems. To combine all these requirements, the scientific community in the field of control system design suggested that model predictive control has a large potential in automotive systems (Ferreau et al., 2007; Shoukry et al., 2010). The biggest challenge derives from the fact that implementation of the predictive control algorithm should be done using finite word length representation because of the limited word length of the control units. In contrast, almost all the automatic control strategies are designed with real numbers and usually implemented in floating point. Moreover, due to truncation errors a stable closed-loop system may become unstable when the infinite precision control law is implemented with a fixed point digital control processors (Song et al., 2001). Further, in terms of control performances, the stability of a digital system may be lost due to the finite word length effects at practical implementation of digital control or filter algorithms (Reddy et al., 2001).

Most traditional control algorithms are based on shift operator description, but these strategies are ill conditioned when applied to data taken at the sampling rates (Vincent et al., 1997). Many algorithms are better conditioned numerically using discrete delta (δ) operator implementation than that ones based on shift (q) operator implementation. The δ-operator is an Euler's approximation to the derivative and this approach has been known as the "difference operator" in the field of numerical analysis. Since the publishing of (Middleton et al., 1986), many researchers paid attention towards identification and adaptive control using δ-operator (Chadwick et al., 2010; Wu et al., 2000). Nowadays, the δ-operator is widely applied in wideband communication and real time digital control (Tatsu et al., 2007; Hongjiu et al., 2010; Dostál et al., 2011; Kerekeset al., 2005).

Relationship between the q-operator and δ-operator has been analyzed in many research papers and it is shown that the δ parameterization has better finite word length (FWL) closed loop stability margin then the traditional discrete time operator. According to Z-Transform, as a sample rate approaches to zero, all stable poles will converge to the point (1, j0). This aspect is related to the increased clustering of the poles and zeros around the point (1, j0) or it is possible that the poles and zeros to go outside the unit circle (Cheng et al., 2007; (Halauca) Budaciu et al., 2008).

Moreover, the q-discrete domain is unconnected with the continuous domain, this is because the continuous domain description cannot be obtained by setting the sampling time $Ts = 0$ (Middleton et al., 1986; Chadwick et al., 2010). There is a close connection between continuous time result and δ representation. Moreover, the δ-operator has certain numerical advantages compared with q shift operator. The paper (Li et al., 1996) compares the δ-operator with the q-operator state space realizations in terms of the effects of the finite word length errors on a transfer function. The results show that the parameterization in the δ-domain provides a superior sensitivity performance over those in the q-operator. The authors Cheng and Chiu in the paper (Cheng et al., 2007) analyze and propose upper bounds on the sample rate, lower frequency of interest and the available word length for controller representation.

Generalized predictive control (Clarke et al., 1987) is the most popular controller among of all predictive control formulations. The shortcomings of these control techniques are particularly emphasized especially when processes are to be run very fast and involve high sampling frequency (Kadirkamanathan et al., 2009).

The present paper proposes a new solution to design a predictive controller for a powertrain system based on the benefits of the δ-operator. The aim of this paper is to show that the SS δGPC algorithm has large potential in the context of fast sampling combine with finite word length digitization. Firstly, the state space model of the automotive powertrain system is converted in the two discrete domains, q and δ representation for different sampling frequencies and it is demonstrated that for 16 bit fixed point representation, the q-domain model has the eigenvalues outside the stable unit circle for fast sampling rate. Further, the SS δGPC algorithm is applied for wheel speed control and compare with the classical q-domain GPC algorithm in the context of small sample period. Many simulation studies have been performed using different predictive control parameters and the results are shown comparatively with the classical GPC controller case. In the final part of this paper, the both algorithms are implemented considering 16 bits fixed point representation and the simulation results are analyzed. The investigation is entitled because nowadays it is more desired the fixed point implementation for electronic control units in automotive industry for the reason of cost, simplicity, power consumption and memory space.

The paper is organized as follows: a brief background to the δ-domain state space model is given in Section 2. The Section 3 deals with the automotive powertrain affine model on a state space approach system and its conversion to the δ-domain. The displacement of the eigenvalues for different sample frequencies is also comparatively analyzed by considering the two discrete state space models. In Section 4, the framework for SS δ GPC is presented. In Section 5, the performance of the equivalent shift and δ-domain predictive control algorithms are compared in the scenario of fast-sampling and finite word length digitization and Section 6 concludes the paper.

2. DELTA DOMAIN STATE SPACE MODEL

The δ-operator can be directly substituted into q-operator from the definition:

$$\delta = \frac{q-1}{T_s} = \frac{e^{sT_s}-1}{T_s}. \tag{1}$$

Although there is a linear transformation between the two discrete domains, the two operators have distinct conceptual roles:

$$q^j = (1+\delta T_s)^j = \sum_{n=0}^{j} C_j^n (T_s\delta)^j, \quad C_j^n = \frac{j!}{n!(j-n)!} \tag{2}$$

where T_s is sampling period and q is the usual forward-shift operator. In the case of continuous time system, the stable poles should be in the left half plane of the complex plane, whereas in the discrete time this becomes the interior of the unit circle. In the case of δ-domain, the stability region increases as the sample rate is higher, so that the stability region is the interior of a circle centred at $(-1/T_s, 0)$ with the radius $1/T_s$ (Middleton et al., 1986). This property gives the delta operator its superior performance at high sample rates as compared to the classical q-operator. The δ discrete state space model is:

$$\begin{aligned} \delta x(k) &= \mathbf{A}_\delta x(k) + \mathbf{B}_\delta u(k) \\ y(k) &= \mathbf{C}_\delta x(k) + \mathbf{D}_\delta u(k) \end{aligned}. \tag{3}$$

The deterministic case of single input single output state space form in the classical representation in the q discrete domain is:

$$\begin{aligned} qx(k) &= \mathbf{A}_q x(k) + \mathbf{B}_q u(k) \\ y(k) &= \mathbf{C}_q x(k) + \mathbf{D}_q u(k) \end{aligned}, \tag{4}$$

where the relationship between the two discrete models is given by ((Halauca) Budaciu et al., 2008):

$$\mathbf{A}_\delta = \frac{\mathbf{A}_q - I}{T_s}, \ \mathbf{B}_\delta = \frac{\mathbf{B}_q}{T_s}, \ \mathbf{C}_\delta = \mathbf{C}_q, \ \mathbf{D}_\delta = \mathbf{D}_q. \tag{5}$$

Middleton and Goodwin have proposed a procedure for obtaining δ state space model directly from the continuous model (Middleton et al., 1986). Thus, they suggest the following relations for conversion:

$$\mathbf{A}_\delta = \frac{e^{\mathbf{A}_c T_s} - I}{T_s} = \Omega \mathbf{A}_c, \ \mathbf{B}_\delta = \Omega \mathbf{B}_c, \ \mathbf{C}_\delta = \mathbf{C}_c, \ \mathbf{D}_\delta = \mathbf{D}_c. \tag{6}$$

where $\mathbf{A}_c, \mathbf{B}_c, \mathbf{C}_c, \mathbf{D}_c$ are continuous-time state space model matrices and:

$$\begin{aligned} \mathbf{\Omega} &= \frac{1}{T_s}\int_0^{T_s} e^{\mathbf{A}_c \tau} d\tau = \frac{1}{T_s}(e^{\mathbf{A}_c T_s} - I)\mathbf{A}_c^{-1} \\ &= I + \frac{\mathbf{A}_c T_s}{2!} + \frac{\mathbf{A}_c^2 T_s^2}{3!} + \ldots\ldots \end{aligned} \tag{7}$$

The correspondence between the two domains is emphasized in the limit case:

$$\begin{aligned} \lim_{T_s \to 0} \mathbf{A}_\delta &= \mathbf{A}_c \\ \lim_{T_s \to 0} \mathbf{B}_\delta &= \mathbf{B}_c \end{aligned}. \tag{8}$$

It is easy to notice that for the limit case $T_s \to 0$, there is no equivalence between the matrices obtained in the q discrete time domain and s domain as one would expect:

$$\lim_{T_s \to 0} \mathbf{A}_q = \mathbf{I}$$
$$\lim_{T_s \to 0} \mathbf{B}_\delta = 0 \quad (9)$$

An advantage of modelling linear systems in the δ-domain consists in the fact that it provides an exact discrete-time representation of the system (Middleton et al., 1986). For a given word length, the all complex values that can be represented are located on fixed grid locations and therefore with finite precision, higher sample rate will not assure an accurate approximation unless the precision is also increased (Clarke et al., 1987). It is to be noted that the highest sample rate is restricted by the resolution of the implementing hardware and the accuracy is also limited by the FWL representation.

3. THE DELTA DOMAIN POWERTRAIN STATE SPACE MODEL

The powertrain system illustrated in Fig.1 consists of the following parts: engine, clutch, transmission (gear box), final drive, driveshaft and wheels (Caruntu et al., 2011). In order to apply the State Space δ GPC algorithm, an accurate powertrain model is required to predict the vehicle's response to a torque input. An affine model with three rotational inertias is used (Caruntu et al., 2011). The state space model with the affine term is converted into the δ-representation. This model will be used to design and simulate the predictive control strategy in the δ-domain.

The powertrain model with three inertias is illustrated in Fig. 1 and the state space model with an affine term developed in (Caruntu et al., 2011; Hrovat et al., 2000) is used.

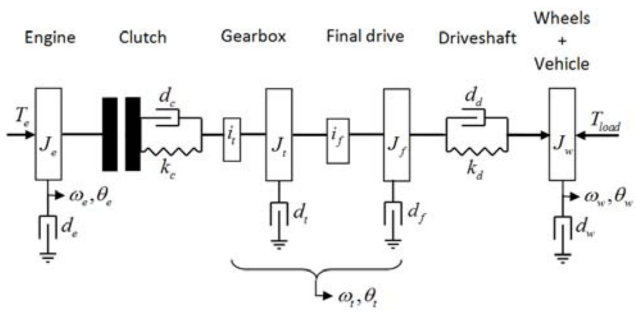


Fig. 1. Three inertias powertrain model.

The state variables of the model considered in this paper are as follows: the torsion angle between engine and transmission defines the first state, the torsion angle between transmission and wheels is the second state and the other three states are: the angular speed of the engine, the angular speed of the transmission and the angular speed of the wheel, respectively. The other parameters of the model are defined in (Caruntu et al., 2011).

$$\begin{aligned} x_1 &= \theta_e - \theta_t i_t \\ x_2 &= \theta_t / i_f - \theta_w \\ x_3 &= \omega_e \\ x_4 &= \omega_t \\ x_5 &= \omega_w. \end{aligned} \quad (10)$$

In this paper the piecewise state space model given in (Caruntu et al., 2011) is described by a collection of sets $\left\{\Omega_i \mid i \in \mathbb{Z}_{[1,4]}\right\}$, with i the active mode at time $t \in \mathbb{R}_+$. The collection of these sets defines a partition of the state space $X \subseteq \mathbb{R}^5$ as follows:

$$\begin{aligned} &\{\Omega_1 := \left\{x \in \mathbb{R}^5 \mid x_3 \le \omega_e^{closing}\right\}, \\ &\{\Omega_2 := \left\{x \in \mathbb{R}^5 \mid x_3 \ge \omega_e^{closing} \ \& \left|x_1\right| \le \theta_1\right\}, \\ &\{\Omega_3 := \left\{x \in \mathbb{R}^5 \mid x_3 > \omega_e^{closing} \ \& \ \theta_1 < \left|x_1\right| \le \theta_2\right\}, \\ &\{\Omega_4 := \left\{x \in \mathbb{R}^5 \mid x_3 > \omega_e^{closing} \ \& \ \theta_2 < \left|x_1\right| \le \theta_3\right\}, \end{aligned} \quad (11)$$

where $\omega_e^{closing}$ is the engine closing speed and θ_i, $i = \overline{1:3}$ are the threshold values for torsion angle between engine and transmission, which are used to pass from one working mode of the clutch to another. In Fig. 2 is illustrated the frequently switches between the operating modes of the clutch considering closed–loop system with horizon-1 predictive controller (Caruntu et al., 2011). This paper addresses the analysis and control design considering the cruise regime in the second operating mode of the clutch. Therefore, the affine state space model uses the parameters values appropriate to the second clutch working mode, see (Caruntu et al., 2011).

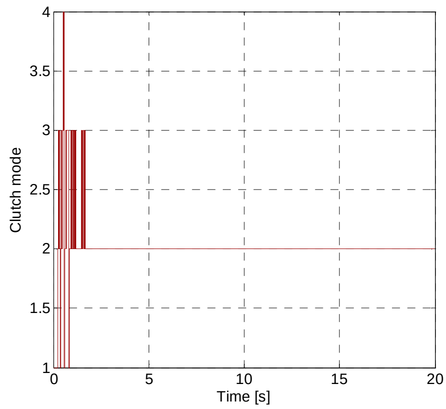


Fig. 2. Switching through the four modes of operation of the clutch.

The affine state-space model in continuous time domain is defined as:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}_c\mathbf{x} + \mathbf{B}_c\mathbf{u} + \mathbf{f}_c \\ \mathbf{y} &= \mathbf{C_c x} \end{aligned} \quad (12)$$

The matrices $\mathbf{A}_c$, $\mathbf{B}_c$, $\mathbf{C_c}$ and the affine term $\mathbf{f}_c$ are as follows:

$$\mathbf{A}_c = \begin{bmatrix} 0 & 0 & 1 & -i_t & 0 \\ 0 & 0 & 0 & \frac{1}{i_f} & -1 \\ -\frac{k_{c2}}{J_1} & 0 & -\frac{d_{c2}+d_e}{J_1} & \frac{d_{c2} i_t}{J_1} & 0 \\ \frac{k_{c2} i_t}{J_2} & -\frac{k_d}{i_f J_2} & \frac{d_{c2} i_t}{J_2} & \frac{-d_{not}}{J_2} & \frac{d_d}{i_f J_2} \\ 0 & \frac{k_d}{J_3} & 0 & \frac{d_d}{i_f J_3} & \frac{-\left(d_w + d_d + c_{r2}\right)}{J_3} \end{bmatrix}, \quad (13)$$

$$\mathbf{B}_c = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{J_1} \\ 0 \\ 0 \end{bmatrix}, \ \mathbf{C_c} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}^T, \ \mathbf{f_c} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -\frac{T_{roll}}{J_3} \end{bmatrix}, \quad (14)$$

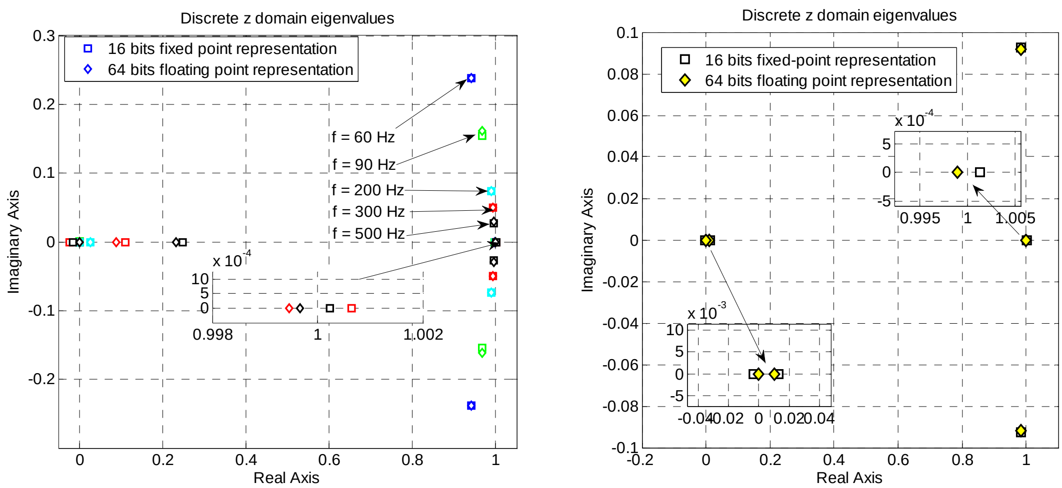
where $d_{not} = \left(d_{c2} i_t^2 + d_2 + \frac{d_d}{i_f^2} \right)$.

The coefficients signification and all the values of the parameters are given in (Caruntu et al., 2011). The input of the system is the engine torque $u = T_e$ and the controlled output is represented by the wheel angular speed.

The simulations were performed in Matlab software, which, by default, employs 64-bits double precision IEEE-754 format for the numbers representation. The calculations were made firstly in floating point, and then the analysis is extended for fixed-point representation, considering 16 bits fixed point word length representation.

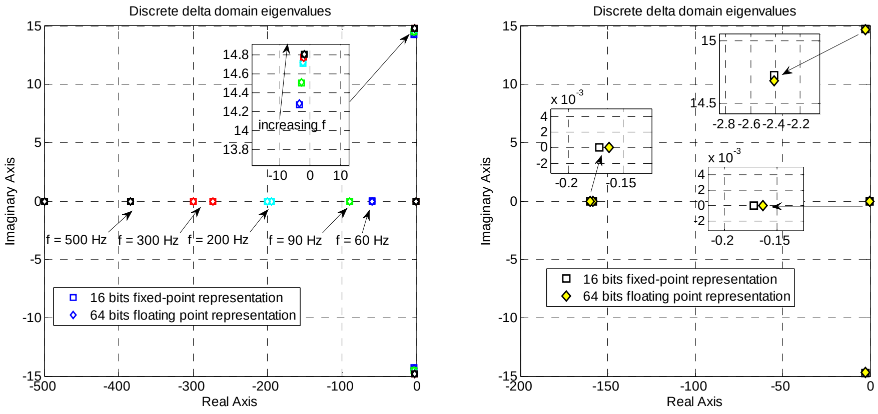
The sampling period of 2ms corresponds to an acceptable compromise between the industrial standard sampling period for such an application (around 10ms) and the corresponding error in terms of velocity and clutch active mode. The higher the sampling rate used, the better is the resolution of the result obtained. Sampling theorem states that the minimum sample frequency required to avoid aliasing needs to be at least twice the highest frequency component, which is also valid in δ-domain. The sampling rate up to 50 times is allowable when using a highly efficient computer (Shim et al., 2006). It is to be noticed that the eigenvalues of the matrix $\mathbf{A}_\delta$ are closed to the values of the $\mathbf{A}_c$ continuous time matrix for $f = 500Hz$ and the values lie in the stability region established by the interior of a circle centered at $(-500, j0)$ with the radius of 500. In fact, the stability region approaches the left half complex plane as the sample rate is higher. However, this approximation cannot be applied when the controller is implemented with finite precision (Song et al., 2011).

The complex values that can be represented are located on fixed grid locations; therefore, with finite precision, higher sample frequency will not provide a more accurate approximation unless the precision is also increased (Cheng et al., 2007). Using functions developed in Matlab, it is possible to obtain quantized values for different word length representation. The function implemented in Matlab for quantization uses rounding operation on sign-magnitude format. In this operation the real number is rounded to the nearest representable level. Because the values contain both integer and fractional part, we need to know the required word length for each component. To estimate the minimum word length both for integer part and fractional part, the relations found in (Cheng et al., 2007) were used. The Matlab function used in this study determines the number of $Binteger$ bits for the magnitude wise largest coefficient of the characteristic equation and Bf bits to the fraction part. Thus, all coefficients have the same $Binteger + Bf + 1$ bit pattern. The eigenvalues for different sampling frequencies are illustrated in Fig. 3 a), b) for discrete z domain and Fig. 3 c), d) for discrete δ-domain, respectively.



a) b)

Fig. 3.1. The eigenvalues displacement for 16 bit fixed-point and 64 bit floating point representation.
a) The discrete z domain eigenvalues for $f = [60,\ 90,\ 200,\ 300,\ 500]Hz$; b) The z discrete eigenvalues for $f = 500Hz$



c) d)

Fig. 3.2. The eigenvalues displacement for 16 bit fixed-point and 64 bit floating point representation.
c) The discrete δ domain eigenvalues for $f = [60,\ 90,\ 200,\ 300,\ 500]Hz$; d) The δ discrete eigenvalues for $f = 500Hz$

Considering the characteristic equation of the system:

$$D_\delta(\delta) = \alpha_n\delta^n + \alpha_{n-1}\delta^{n-1} + \ldots + \alpha_1\delta + \alpha_0\,, \tag{15}$$

with λ_i being the roots of the equation (15). According with (Ingle et al., 2007), number representation is in the range of $\overline{0:\ (2^N - 1)/2^{Bf} = 2^{Binteger} - 2^{-Bf}}$. The required word length for the fractional part of the j^{th} root can be derived as (Cheng et al., 2007):

$$Bf_j = \left\lfloor \log_2\left(\frac{\left|\sum_{k=1}^{n}\lambda_j^k\right|}{\prod_{k=1,k\neq j}^{n}(\lambda_j - \lambda_k)}\right)\cdot\frac{1}{\varepsilon_j}\right\rfloor + 1\,. \tag{16}$$

Therefore, the required word length of the fractional part can be derived by taking the maximum word length among all poles:

$$Bf = \max_j\{Bf_j\}\ . \tag{17}$$

The deviation ε_j is defined in (Cheng et al., 2007) as the expected change in the location of the roots of the characteristic equation accordingly with the changes in the characteristic equation coefficients. The coefficients variations are the results of round-off when implementing in a fixed point processor with finite word length. The required word length for the representation of the integer part is defined as:

$$Binteger = \max_i\{\lfloor\log_2(\alpha_i)\rfloor\} + 1\ . \tag{18}$$

Further, the eigenvalues of the system described above are represented based on this coupled relationship to estimate the necessary resolution. Considering the deviation of the eigenvalues of value 0.01 and 16 bits fixed point

representation, the variation of the eigenvalues are illustrated in Fig. 3.

The results from Fig. 3 a) show that, the eigenvalues obtained in the classical discrete time domain for fast sampling cluster the point (1, j0) and one of them becomes unstable for 16 bits fixed point representation as is expanded in Fig. 3 b). Hence, the figures show that the δ-operator performs consistently as sample rate is increased as is seen in Fig. 3 c), d), but that the accuracy of the q-operator degrades as sample rate increases. In the following, the δ state space formulation analyzed in this paragraph will be used to calculate the predictions for the predictive control algorithm formulated in the δ-domain.

4. STATE SPACE APPROACH OF THE GPC DESIGN IN δ-DOMAIN

Considering the deterministic case of single input single output δ-domain state space model with the known states unaffected by disturbance or noise (Song et al., 2011):

$$\begin{aligned}\delta\mathbf{x}_k &= \mathbf{A}_\delta\mathbf{x}_k + \mathbf{B}_\delta\mathbf{u}_k\\ \mathbf{y}_k &= \mathbf{C}_\delta\mathbf{x}_k\end{aligned}\ , \tag{19}$$

with $\mathbf{x}_k \in \mathbb{R}^n, \mathbf{u}_k \in \mathbb{R}^m, \mathbf{y}_k \in \mathbb{R}^p$ the state vector, the control vector and the output vector, respectively. Proceed from this model, the j-$^{\text{th}}$ order δ derivatives state are obtained:

$$\delta^j x_k = \mathbf{A}_\delta{}^j x_k + \sum_{i=0}^{j-1}\mathbf{A}_\delta{}^{j-i-1}\mathbf{B}_\delta\delta^i u_k\ , \tag{20}$$

with $j = \overline{0, N_y}$, N_y being the prediction horizon. Using the model (19) the following δ derivative predictors are:

$$\delta^j y_k = \mathbf{C}_\delta\mathbf{A}_\delta{}^j x_k + \sum_{i=0}^{\min\{j,Ny\}-1}\mathbf{C}_\delta A_\delta{}^{j-i-1}\mathbf{B}_\delta\delta^i u_k\,. \tag{21}$$

In a matrix notation the expression of δ derivatives predictors are:

$$\hat{\mathbf{y}}_\delta = \mathbf{f} + \mathbf{G}\mathbf{u}_\delta\,, \tag{22}$$

where

$\mathbf{u}_\delta = [u_k\quad \delta u_k\quad \delta^2\, u_k \ldots\ldots \delta^{N_u-1}u_k]^T$,

$\hat{\mathbf{y}}_\delta = [y_k\quad \delta y_k\quad \delta^2\, y_k \ldots\ldots \delta^{N_y}\, y_k]^T$.

$\mathbf{G}$ is the expanded Toeplitz matrix containing the δ based Markov parameters and it has the dimension $(N_y+1)\times N_u$:

$$\mathbf{G}=\begin{bmatrix} g(0,0) & \dots & g(0,N_u-1) \\ \vdots & \ddots & \vdots \\ g(N_y,0) & \dots & g(N_{y,}N_u-1) \end{bmatrix}, \tag{23}$$

where

$$g(j,i)=\begin{cases}\mathbf{C}_\delta \mathbf{A}_\delta{}^{j-i-1}\mathbf{B}_\delta , 0\le i\le \min\{k,N_u\}-1 \\ 0,\ \text{otherwise},\end{cases} \tag{24}$$

and $\mathbf{f}$ the free response:

$$\mathbf{f}=\left[\mathbf{C}_\delta \mathbf{A}_\delta{}^{\mathbf{1}} \quad \dots \quad \mathbf{C}_\delta \mathbf{A}_\delta{}^{\mathbf{Ny}}\right]^{\mathrm{T}}\mathbf{x}_k . \tag{25}$$

The optimal control sequence is obtained by minimizing an objective function, knowing the reference trajectory r_{k+i}:

$$J=\sum_{i=N_1}^{N_y}\left[\hat{\mathbf{y}}_{k+i}-\mathbf{r}_{k+i}\right]^2+\lambda\sum_{i=1}^{N_u}\left[\Delta\mathbf{u}_{k+i-1}\right]^2, \tag{26}$$

where N_u is the control horizon, N_1 is minimum costing horizon and λ is control weighting factor. In order to obtain the optimal control sequence in the δ-domain, the set of vectors that arise in criterion function are obtained from mapping the q-domain terms into the δ-domain through binomial expansion (2). The control increments are preferred instead of absolute values to avoid steady state error. In the classical discrete time domain the control increments are subject to:

$$\Delta u_{k+j}=u_{k+j}-u_{k+j-1}=0, \quad \mathrm{j}>N_u . \tag{27}$$

The constraints on control increments are obtained (Kadirkamanathan et al., 2009):

$$\begin{bmatrix}\Delta u_k \\ \Delta u_{k+1} \\ \vdots \\ \Delta u_{k+N_u+1}\end{bmatrix}=\underbrace{\begin{bmatrix}1 & 0 & 0 & \cdots & 0 \\ -1 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 1 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & -1 & 1\end{bmatrix}}_{\mathbf{Q}}\mathbf{K}_u\begin{bmatrix}\delta^0 u_k \\ \delta u_k \\ \delta^2 u_k \\ \vdots \\ \delta^{N_u}u_k\end{bmatrix}+\begin{bmatrix}-u_{k-1} \\ 0 \\ \vdots \\ 0\end{bmatrix}. \tag{28}$$

where $\mathbf{K}_u$ is the matrix obtained from the conversion (2) analogous with $\mathbf{K}_y$.

$$\begin{bmatrix}q^0 y_k \\ q^1 y_k \\ \vdots \\ q^j y_k \\ \vdots \\ q^{N_y} y_k\end{bmatrix}=\underbrace{\begin{bmatrix} T_s{}^0 & 0 & 0 & \dots & 0 \\ T_s{}^0 & 1T_s{}^1 & 0 & \dots & 0 \\ T_s{}^0 & 2T_s{}^1 & T_s{}^2 & \dots & \vdots \\ \vdots & \vdots & \vdots & \dots & \\ T_s{}^0 & \frac{j!}{(j-1)!}T_s{}^1 & \dots & \dots & \frac{j!}{(j-i)!i!}T_s{}^i \\ \binom{N_y-1}{0}T_s{}^0 & \binom{N_y-1}{1}T_s{}^1 & \binom{N_y-1}{2}T_s{}^2 & \dots & \binom{N_y-1}{N_y-1}T_s{}^{N_y-1}\end{bmatrix}}_{\mathbf{K}_y}\begin{bmatrix}\delta^0 y_k \\ \delta^1 y_k \\ \vdots \\ \delta^j y_k \\ \vdots \\ \delta^{N_y} y_k\end{bmatrix} \tag{29}$$

The relation between the two discrete domains with respect to constraint (28) is easy to obtain using two components of: $\mathbf{u}_\delta=\left[\mathbf{u}_\delta^1 \quad \mathbf{u}_\delta^2\right]^T$

$$\begin{aligned}\mathbf{u}_\delta^1&=[\delta^0 u \qquad \delta^1 u \qquad \dots\delta^{N_u}u]^T \\ \mathbf{u}_\delta^2&=[\delta^{N_u+1}u \quad \delta^{N_u+2}u \quad \dots\delta^{N_y-1}u]^T .\end{aligned} \tag{30}$$

The control increments can be expressed as (Kadirkamanathan et al., 2009):

$$\Delta\mathbf{u}_{k+i-1}=\mathbf{Q}\mathbf{K}_u\mathbf{u}_\delta^1,\ i=\overline{1:N_u} . \tag{31}$$

Once arrived at the control horizon value, no adjustment can be done in control sequence, so that the command action is kept constant on the entire left interval. Therefore it is necessary to impose the condition:

$$u_{k+i}=u_{k+N_u},\ i=\overline{N_u+1,N_y-1} \tag{32}$$

or

$$\begin{bmatrix}u_{k+N_u+1} \\ u_{k+N_u+2} \\ \vdots \\ u_{k+N_y-1}\end{bmatrix}=\begin{bmatrix}u_{k+N_u} \\ u_{k+N_u} \\ \vdots \\ u_{k+N_u}\end{bmatrix}=\vec{\mathbf{\Gamma}}\cdot\mathbf{u}_\delta^1 , \tag{33}$$

with $\vec{\mathbf{\Gamma}}=\left[\binom{N_u}{0}T_s{}^0 \quad \binom{N_u}{1}T_s{}^1 \quad \dots \quad \binom{N_u}{N_u}T_s{}^{N_u}\right]$.

From (30) and (32) one may obtain:

$$\mathbf{\Gamma}_u\mathbf{u}_\delta^1+\mathbf{\Gamma}_y\mathbf{u}_\delta^2=\vec{\mathbf{\Gamma}}\mathbf{u}_\delta^1 , \tag{34}$$

where $\mathbf{\Gamma}_u$, $\mathbf{\Gamma}_y$ are sub matrices of $\mathbf{K}_u$ with the number of columns equals the dimensions of $\mathbf{u}_\delta^1$ and $\mathbf{u}_\delta^2$, respectively so that:

$$\mathbf{u}_\delta^2=\mathbf{\Gamma}_y{}^{-1}(\vec{\mathbf{\Gamma}}-\mathbf{\Gamma}_u)\mathbf{u}_\delta^1 . \tag{35}$$

The cost function has an important role in designing predictive control strategies and in the δ-domain can be expressed similarly (Kadirkamanathan et al., 2009). The predictor expression (21) is finally rewritten:

$$\hat{\mathbf{y}}_\delta=\mathbf{f}+\left[\mathbf{G}_u+\mathbf{G}_y\mathbf{\Gamma}_y{}^{-1}(\vec{\mathbf{\Gamma}}-\mathbf{\Gamma}_u)\right]\mathbf{u}_\delta^1 , \tag{36}$$

with

$$\mathbf{G}_u\in\mathbb{R}^{n(N_y+1)\times m(N_u+1)} \text{ and } \mathbf{G}_y\in\mathbb{R}^{n(N_y+1)\times m(N_y-N_u-1)} \tag{37}$$

Considering (35) and the predictor (37) into the criterion function (26) and differentiating with respect to $\mathbf{u}_\delta^1$ produces the optimal strategy:

$$\frac{dJ_\delta}{d\mathbf{u}_\delta^1}=0\Rightarrow \quad \mathbf{u}_\delta^1=-(\mathbf{H}^T)^{-1}\mathbf{L}. \tag{38}$$

where:

$$\begin{aligned}&\mathbf{f}_0=(\mathbf{f}-\mathbf{w})^T\mathbf{K}_y{}^T\mathbf{K}_y(\mathbf{f}-\mathbf{w}), \\ &\tilde{\mathbf{G}}=\mathbf{G}_u+\mathbf{G}_y\mathbf{\Gamma}_y{}^{-1}(\vec{\mathbf{\Gamma}}-\mathbf{\Gamma}_u), \\ &\mathbf{H}=\tilde{\mathbf{G}}^T\mathbf{K}_y{}^T\mathbf{K}_y\tilde{\mathbf{G}}+\lambda\mathbf{K}_u{}^T\mathbf{Q}^T\mathbf{Q}\mathbf{K}_u, \\ &\mathbf{L}=\tilde{\mathbf{G}}^T\mathbf{K}_y{}^T\mathbf{K}_y(\mathbf{f}-\mathbf{w}).\end{aligned} \tag{39}$$

The reference vector is assumed to be of the form:

$$\mathbf{w}_\delta=\left[\delta^0 w \quad \delta^1 w \quad \delta^2 w\dots\dots \delta^{N_y}w\right]^T . \tag{40}$$

The State Space δ GPC algorithm is remarkable by the fact that the strategy for determining the optimal control is fully developed in the δ-domain. Therefore, the rounding errors that occur in the classic q-domain are reduced in this case, especially for small sampling periods in the context of finite number of bits representation as it will be illustrated in the following.

5. SIMULATION RESULTS

This section presents the performances of the GPC strategy designed in the δ-domain, investigated on the automotive powertrain model considering finite word length representation. The simulation experiments were carried out in Matlab, firstly in floating point, and then the analysis will be extended for fixed-point representation.

5.1 Simulation results using 64 bits floating point

The controller has to satisfy some physical constraints on input and states, so the engine torque is restricted by lower and upper bounds (Caruntu et al., 2011):

$$\begin{aligned} &0 \le u_k \le T_e^{\max}, \\ &-T_e^{\Delta} \le \Delta u_k \le T_e^{\Delta}, \end{aligned} \tag{41}$$

where $T_e^{\max}$ is the maximum torque that can be generated by the internal combustion engine, $\Delta u_k = u_k - u_{k-1}$, for all $k \in \mathbb{Z}_{\ge 1}$ and T_e^{Δ} is the maximum increase or decrease in torque at each sampling instant. In the simulations, these constraints were set to $T_e^{\max} = 200$ Nm and $T_e^{\Delta} = 3$. Fig. 4 illustrates the comparisons of engine torque and wheel speed using both SS δ GPC and classical GPC strategy. The control parameters were set as follows: sample frequency $f = 300Hz$, the control horizon is fixed to $N_u = 1$, and the prediction horizon is fixed to $N_y = 8$; the control weighting parameter is $\lambda = 0.0002$ in each case. The engine torque represented in the first figure is referred as input signal and the output, the wheel speed which must follow the reference trajectory drawn with dotted line, in the second plot. As it is shown, the SS δ GPC performs much smoother than the classical GPC algorithm. The wheel speed reaches the desired value with very small oscillations as is shown with solid line in the second plot of Fig.4. The performances of algorithm are shown further in Fig. 5, when varying the control horizon, for $\lambda = 0.2$ subject to fast sample rate of $f = 300Hz$ .The system response is demonstrable sensitive to the variation in control weighting factor.

The SS δ GPC algorithm is able to offer better performance via manipulation of prediction horizon N_y and weighting factor λ. However, the aim of this study is to highlight the performances of the proposed algorithm for fast dynamic results under a fast sampling frequency.

An important point to note in predictive control algorithms refers to the complexity of computing and computing time. Another advantage of working in the δ-domain is that numerical properties are typically improved at fast sample rate.

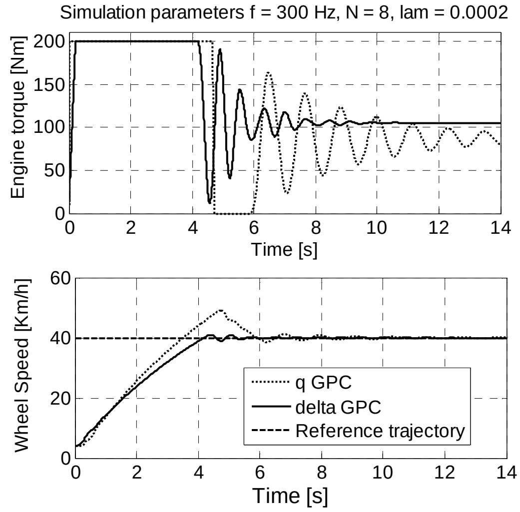


Fig. 4. Comparison of control signal and system output using both δ and q GPC strategy for fast sample rate.

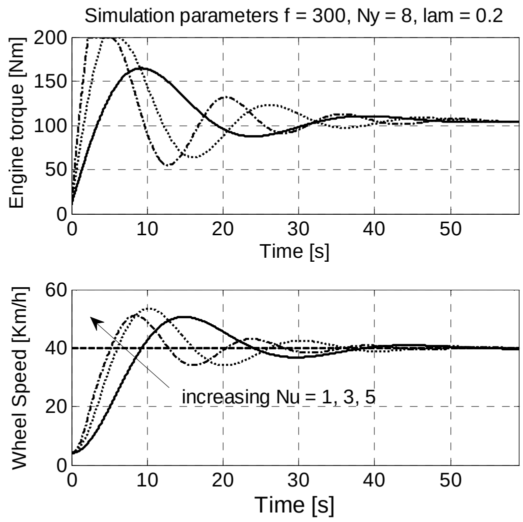


Fig. 5. Performance of SS δ GPC algorithm for varying values of control horizon for sample frequency of $f = 300Hz$.

This feature can be investigated by analyzing Fig. 6 where the simulation results are plotted for both algorithms in the context of fast sampling, when prediction horizon varies.
In the simulations, the sample rate was fixed to $f = 300Hz$; the weighting factor is kept constant at the value of $\lambda = 0.0002$ and the control horizon $N_u = 1$. Although, the values for prediction horizon vary very little from $N_y = 6$ to $N_y = 7$, in the case of classical q-GPC algorithm, the output varies significantly as is shown in the second plot of Fig 6 with dotted line (bold dotted line for $N_y = 6$ and thin dotted line for $N_y = 7$).

Not the same thing happens for the case of SS δ GPC, where the evolution of the wheel speed is close related. Moreover, a lower value of prediction horizon implies a lower computational complexity and lower computing time, especially when the algorithm is implemented using finite word length to perform numerical calculations.

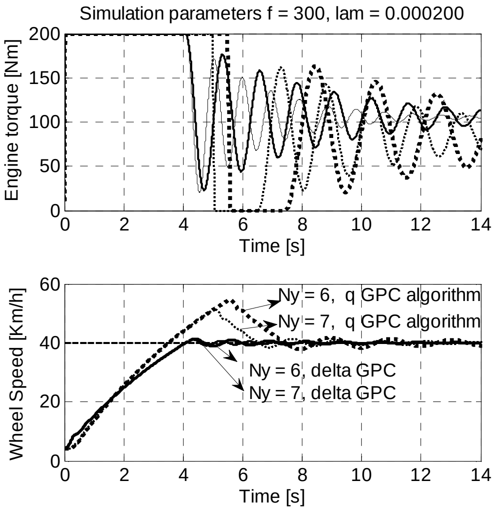


Fig. 6. Comparison between performances of SS δ GPC algorithm and q-domain GPC for varying values of prediction horizon.

5.2 Simulation results using 16 bits fixed point

Control algorithms designed in discrete q-domain become numerically unstable in high-speed sampling making overall control unstable (Tatsu et al., 2007; Song et al., 2011; Kherji et al., 2011). In the following, the two discrete predictive control strategies are quantized using 16 bits word length. The classical discrete time GPC strategy easily becomes numerically unstable, because the word length is too short to express small values. This is due to the fact that at each sampling instant the difference between calculations becomes extremely small. If one still prefers to design the classical GPC algorithm using the q discrete time parametric model obtained from the realization (12) and represent it using 16 bits, achieve the impossibility to represent polynomials containing very small amounts. If the coefficients are quantized using 16 bits fixed point according to coefficient quantization method (Ingle et al., 2007) the range of number representation is from 0 to $(2^N-1)/2^{Bf}=2^{Binteger}-2^{-Bf}$ and all coefficients smaller than $2^{-15}=3.0518\cdot10^{-5}$ will be truncated to 0. Therefore, with the polynomials obtained for 16 bits implementation, the classical q-domain GPC strategy cannot be implemented. Moreover, most of the studies for the implementation of the predictive controller do not use microcontrollers because of the huge amount of time needed to implement this method especially when are constraints. However, by proposing an optimized algorithm with reduced size of code combined with a low cost microcontroller (Kherji et al., 2011) it is possible to obtain an efficient GPC controller based on δ operator. The SS δ GPC calculates the optimal control action based on the matrices of the state space model and the minimum word length for matrices representation is 16 bits including the sign bit (the required number of bits for each matrix is shown in Table 1). It should be mentioned that the minimum number of bits required for fractional part are important only for $\mathbf{B}_\delta$ matrix, because it contains small values.

Table 1

Matrix	Integer bits	Fractional bits
$\mathbf{A}_\delta$	13	2
$\mathbf{B}_\delta$	3	12
$\mathbf{C}_\delta$	1	14

Fig. 7 shows comparatively the evolutions of control action and system output for both algorithms, considering sampling time of $T_s=10ms$ in the context of the same conditions and constraints.

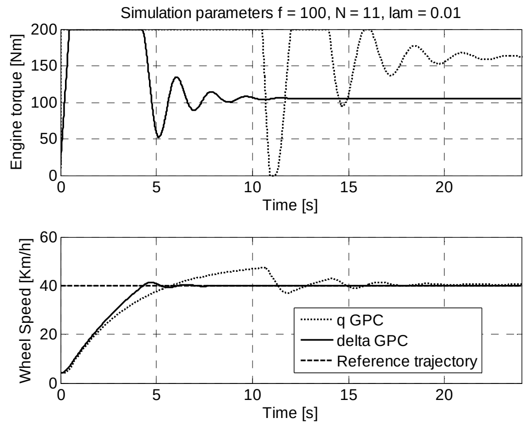


Fig. 7. Comparison results for sampling frequency of $f=100Hz$ using 16 bits fixed point.

In this case, the prediction horizon is increased to the value of $N_y=11$ and the weighting factor is fixed to $\lambda=0.01$. We consider that the quantities $\mathbf{H},\mathbf{G},\mathbf{K}_y,\mathbf{K}_u$ from the optimal control action (38) can be computed off line using floating point accuracy before implementation in a control system.

Quantization introduces noise into the closed-loop system. Moreover, it can affect stability of the system because of the unwanted modification of the position of the poles. Unfortunately, GPC algorithm using shift operator requires a long prediction horizon and a great value for λ to obtain a response without oscillations.

6. CONCLUSIONS

The main objective of this paper is directed to those applications for which the implementation of predictive control strategy on finite word length digital hardware would bring both numerical and conceptual advantages in terms of control performances. A new solution proposed in this paper is to design a predictive control strategy in the discrete δ-domain. State Space δ Generalized Predictive Control can bring significant improvements, especially when it is implemented on embedded systems in the context of fast sampling. In this paper, the predictive controller design in δ-domain was successfully applied on a powertrain system model. The paper analyzes the performances of the proposed solution in comparison with the equivalent classical GPC algorithm for different sampling frequencies. The paper demonstrates that the new solution proposed for wheel speed control of a powertrain system performs better than the classical GPC controller at high sampling rate and is also able to offer lower computational complexity due to small prediction horizon. Moreover, SS δ GPC exhibits at high sampling rates better numerical conditioning due to its less sensibility to the coefficients rounding errors at low precision.

ACKNOWLEDGEMENTS

The first author acknowledges the support of the CNCS-UEFISCDI grant PN-II-RU PD 331/2010.

REFERENCES

(Halauca) Budaciu, C., Lazăr, C., 'Delta Domain Predictive Control for Fast, Unstable and Non-minimum Phase Systems', *Control Engineering and Applied Informatics*, 2008, 10, (4), pp. 26 - 31.

Caruntu, C. F., Balau, A. E., Lazar, M., P. P. J. van den Bosch and Cairano, S. Di., 'A Predictive Control Solution for Driveline Oscillations Damping'. *In: Hybrid Systems: Computation and Control*, Chicago, USA, 2011, pp. 181–190.

Chadwick, M. A., Anderson, S. R., Kadirkamanathan, V.,'An iterative Kalman smoother/least-squares algorithm for the identification of delta-ARX models', *International Journal of Systems Science*, 2010, 41, (7), pp. 839 – 851.

Cheng, H-M. and Chiu, G. T., 'Coupling between Sample Rate and Required Wordlength for finite Precision Controller Implementation with Delta Transform', *Proceedings of the American Control Conference*, New York, July 11-13, 2007.

Cheng, H-M. and Chiu, G. T., 'Wordlength Estimation of Digital Controller Synthesis for Inkjet Printer Mechanism', *Journal of Computers*, 2008, 3, (4), pp. 50-57.

Clarke, D.W., Mohthadi, C., Tuffs, P.S., 'Generalised predictive control-Part I. The basic algorithm, Part II. Extensions and Interpretations', *Automatica*, 1987, 23, pp. 137–160.

Dostál, P., Vojtěšek, J., Bobál, V., and Babík, Z., '2DOF adaptive control of a tubular chemical reactor', Recent Researches in Circuits, Systems and Signal Processing, *Corfu Island Greece*, July 2011.

Ferreau, H. J., Ortner, P., Langthaler, P., Luigi del Re, Diehl, M., 'Predictive Control of a Real-World Diesel Engine using an Extended Online Active Set Strategy', *Annual Reviews in Control,* 2007.

Hongjiu, Y, Yuanqing, Xia, Mengyin, Fu and Peng, Shi, 'Robust Adaptive Sliding Mode Control for Uncertain Delta Operator Systems', *International Journal of Adaptive Control and Signal Processing*, 2010, 24, pp. 623–632.

Hrovat, D., Asgari, J. and Fodor, M., 'Automotive mechatronic systems', (Mechatronic Systems Techniques and Applications, Vol. 2, *Gordon and Breach Science Publishers*, Inc., 2000).

Ingle V. K. and Proakis J.G., 'Digital Signal Processing using Matlab', (*International Student Edition, Nelson a division of Thomson*, 2007).

Kadirkamanathan, V., (Halauca) Budaciu, C. and Anderson, S., 'Predictive control of fast-sampled systems using the delta-operator', *International Journal of Systems Science*, 2009, 40 (7), pp. 745 – 756.

Kerekes, S. D., Lungeanu, T.; Nakhost, M., Teodorescu, P., Andersen, R., Liserre, G.K., 'Low-Cost Digital Implementation of Proportional-Resonant Current Controllers for PV Inverter Applications using Delta Operator', 31st *Annual Conference of IEEE Industrial Electronics Society*, 2005.

Kherji, A., Bouani, F., Ksouri, M. and Ben Ahmed, M., 'A Microcontroller Implementation of Constrained Model Predictive Control', *International Journal of Electrical and Electronics Engineering*, 2011, pp. 272-279.

Li, G. and Gevers, M., 'On the structure of digital controllers in sampled data systems with FWL consideration', *Proceedings of the 35th IEEE Conference on Decision and Control*, Japan, 1996, pp. 919 - 920.

Middleton, R., Goodwin, G., 'Improved finite word length characteristics in digital control using delta operators', *IEEE Transactions on Automatic Control*, 1986, 31, pp. 1015 - 1021.

Reddy, H.C., Hung, K. and Moschtyz, G.S., 'Realization of Sampled Data Filters with Infinite Transmission Zeros Using Delta Operator', *European Conference on Circuit Theory and Design*, Espoo, Finland, August 2001.

Shim, K-H. and. Sawan, M. E, 'Singularly perturbed unified time systems with low sensitivity to model reduction using delta operators', *International Journal of Systems Science*, 2006, 37 (4), Taylor & Francis, Inc. Bristol, PA, USA.

Shoukry, Y., Watheq, M., Kharashi El and Sherif Hammad, 'MPC-on-Chip: An Embedded GPC Co-processor for Automotive Active Suspension System', *IEEE Embedded Systems Letters*, 2010, 2 (2), pp. 31-34.

Song, T. and Collins, E. G., 'Improved Closed Loop Stability for Fixed Point Controller Realizations using the Delta Operator', *International Journal of Robust and Nonlinear Control*, 2001, 11, pp. 41 – 57.

Song, T., Collins, E. G. and Istepanian, R H., 'Improved Closed Loop Stability for fixed point Controller Realizations using Delta Operator', *International Journal of Robust and Nonlinear Control*, 2011, 11, pp. 41 – 57.

Tatsu, Aoki, 'Implementation of Fixed-Point Control Algorithms Based on the Modified Delta Operator and Form for Intelligent Systems', *Journal of Advanced Computational Intelligence and Intelligent Informatics*,2007, 11, (6), pp. 709-714.

Vincent, Poor H., 'Delta-Operator Based Signal Processing: Fast Algorithms for Rapidly Sampled *Data' Proceedings of the 36th Conference on Decision & Control* San Diego, California USA, December 1997.

Wu, J., Chen, S., Li, G., Istepanian, R.H., Chu, J., 'Shift and Delta Operator Realizations for Digital Controllers with Finite-Word-Length Considerations', *IEE Proc., Control Theory Applications*, 2000, 147(6), pp. 664-672.