International Journal of Dynamics and Control (2019) 7:452–468
https://doi.org/10.1007/s40435-018-0461-8

Stability analysis and efficiency of EMPC for Type-1 systems

M. A. Aravind[1] · Niranjan Saikumar[2] · N. S. Dinesh[3] · K. Rajanna[1]

Received: 15 March 2018 / Revised: 4 May 2018 / Accepted: 11 July 2018 / Published online: 19 July 2018

Abstract
Experience mapping based predictive controller (EMPC) is a recently developed controller based on the concepts of Human Motor Control. It has been demonstrated to out-perform other classical controllers like proportional-derivative (PD), model reference based adaptive controller (MRAC), linear quadratic regulator (LQR) and the linear quadratic Gaussian (LQG) for both Type-1 and Type-0 systems. This paper analyses the stability and efficiency of EMPC for Type 1 systems. EMPC uses rectangular pulse input as control action for well-damped Type 1 systems and a first order decay input for under-damped Type 1 systems . The simulation results of EMPC for position control of a DC motor with a load coupled through a flexible shaft are presented as a case study to derive and prove the stability criterion. The efficiency of EMPC on a practical system is analysed in terms of energy dissipated in the armature resistance of the motor and the same is compared with PD, MRAC, LQR, LQG controller. Further, the computational cost of EMPC is discussed and compared with traditional controllers from the point of view of implementation.

Keywords Optimal control system · Experience mapping based predictive controller · Position control · Flexible shaft · DC motor · Stability · Efficiency · Computational cost

1 Introduction

Stability analysis of controllers is a important requirement to prove the usage of the controller in various applications.

Linear control theory use tools like root locus, Nyquist or Bode plots to prove the stability of the controller. In modern control systems, digital controller algorithms like linear quadratic regulator (LQR), linear quadratic Gaussian (LQG), model reference adaptive controller (MRAC) [1], sliding mode control (SMC), fuzzy control [2] and neural networks [3] use more complex theory like linear matrix inequalities (LMI) based on Lyapunov's stability criterion [4,5] to prove stability. These controllers require the plant model and a set of state space equations that govern the system transfer function [6]. Complex systems such as those involving flexible shafts, have complicated equations and involve numerous state variables that need to be measured. Designing a control system which is stable for an open loop under-damped system or a non-minimum phase system or one that have many poles or zeros is a difficult task. The cost of the implementation of such digital controllers is high.

In literature, certain controllers are also modified to improve stability by using feed-forward control [7]. When the system to be controlled consists of higher order poles, PID controller will be the easiest to implement by using simple techniques like pole-placement based on the required performance criteria as opposed to state based controllers which require higher order matrices to be constructed. PID has also been improved upon by controlling the PID gains using modified LQR or LQG controllers [8,9] to give better response to noise and disturbance. To adapt these controllers for change in system parameters, estimators like an extended Kalman Filter is used so that the system does not become unstable for a greater range of parameter changes [10,11].

✉ M. A. Aravind
aravindm@iisc.ac.in

Niranjan Saikumar
n.saikumar@tudelft.nl

N. S. Dinesh
dinesh@iisc.ac.in

K. Rajanna
kraj@iap.iisc.ernet.in

[1] Department of Instrumentation and Applied Physics, Indian Institute of Science, Bangalore, India

[2] Precision and Microsystems Engineering 3ME, TU Delft, Delft, The Netherlands

[3] Department of Electronic Systems Engineering, Indian Institute of Science, Bangalore, India

In the case of advanced non-linear controllers applied on non-minimum phase systems, equations for stability are derived from Lyapunov-Krasovskii and applied on Fuzzy based [12] and Neural network based [13] controllers. Thus, design of a controller for a given system requires the knowledge of the stability criterion so that we can be confident that the closed-loop system will remain stable for predefined conditions.

Recently, a new control algorithm called the experience mapping based predictive controller (EMPC) was developed for position control [14–17] and speed control [18,19] of a DC motor and shown to outperform other robust controllers. The concepts were further improved to extend the control for under-damped Type-1 systems in [20,21].

1.1 Experience mapping based predictive control (EMPC)

EMPC is inspired from human motor control (HMC). Humans develop skills by practice and interaction with the surrounding environment [22,23]. They can adapt to changes in the environment and improve their motor control skills. The experiences of these interactions are mapped in the brain as input-output relationships. Repeated interaction results in improved experiences, which are stored as motor memories. These motor memories allow the HMC to operate in quasi-open loop, by providing the necessary action to achieve a desired response, based on past experiences [24,25].

EMPC closely follows the following principles. During the learning phase, the system steady state outputs are mapped to the applied inputs and stored in an experience mapped knowledge base (EMK). For a given demand, the control action required is then evaluated from the EMK and applied to the system.

EMPC is capable of handling minor errors introduced due to sensor or system noise and disturbances through the use of Iterative Predictive Action. These errors can also occur due to small changes in the system parameters. However, significant changes in system parameters can result in large errors, oscillations and even instability. Stability analysis of controllers is one of the important aspects of controls theory and various approaches have been used for the same with different controllers. Hence, it is important to analyze and establish the stability conditions of EMPC under changing system conditions.

In this paper, we introduce the basics of EMPC based on earlier literature and build upon the stability criteria for the same. Section 2 derives the stability criteria for EMPC applied for a well-damped Type-1 system. In Sect. 3, we derive the stability criteria for EMPC applied to an under-damped Type-1 system. Section 4 compares EMPC with PD controller for stability. Section 5 the efficiency of EMPC with PD, MRAC and LQG when applied to position control of a DC motor connected to a load through a flexible shaft based on the energy dissipated by the armature resistance. Section 6 compares the efficiency when non-linearities like dry friction are introduced into the motor-load system. In Sect. 7, computational cost and controller memory usage of EMPC is compared with other controllers practical implementation.

2 EMPC for the well-damped Type 1 system

EMPC has been developed for a well damped n^{th} order Type 1 system and implemented on a PMDC motor position control system [15]. The general form of the well-damped Type 1 system is described by Eq. 1

$$G(s) = \frac{(s+z_1)(s+z_2)\cdots(s+z_m)}{s(s+p_1)(s+p_2)\cdots(s+p_n)} \tag{1}$$

where $(z_i, p_j) \in \Re$, $i \in [1,m]$, $j \in [1,n]$, $m \leq n$ for a causal system. EMPC proposes a control action consisting of a rectangular pulse of fixed amplitude with varying pulse-width duration which are mapped to corresponding steady state responses using an EMK [14]. The output $Y(s)$ of the system $G(s)$ when a rectangular input of amplitude A_m and time width T_{on} is applied is defined by Eq. 2

$$Y(s) = A_m \frac{\left(1 - e^{(-sT_{on})}\right)}{s} G(s) \tag{2}$$

The steady state response for the rectangular input using Final Value Theorem is given by Eq. 3

$$Y_{ss} = \frac{b_0}{a_0} A_m T_{on} \tag{3}$$

where $b_0 = \prod_{i=1}^{m}(-z_i)$ and $a_0 = \prod_{i=1}^{n}(-p_i)$. From Eq. 3, it is seen that the steady state output of the system $G(s)$ is reached by applying a rectangular input of width T_{on} from fixed steady state initial condition. The steady state value Y_{ss} is directly proportional to T_{on} if the parameters that define the system (a_0, b_0, A_m) are constant. Therefore Eq. 3 is simplified as Eq. 4

$$Y_{ss} = K_{sa} T_{on} \tag{4}$$

where $K_{sa} = \frac{b_0}{a_0} A_m$ is the constant of proportionality for the given system determined by $G(s)$. A_m is considered to be constant since the input to the system can be considered to be normalized to have an maximum amplitude of 1, which is then amplified by some drive circuitry by a value of Am and applied to the system $G(s)$. Figure 1 shows the response of a Type 1 system for the application of a standard rectangular input from steady state.

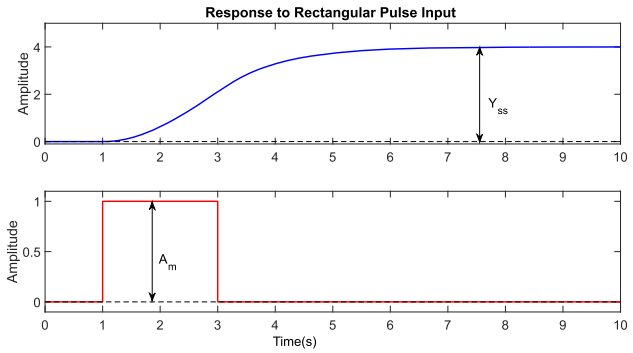

Fig. 1 Standard rectangular input action of EMPC and response of a Type 1 system $G(s) = \frac{s+10}{s(s^2+6s+5)}$ with $A_m = 1$, $K_{sa} = 2$ and $T_{on} = 2$

Equation 4 establishes that there is a linear relation between the parameter T_{on} and the steady state response Y_{ss} for a fixed plant defined by $G(s)$. For a given Demand D, EMPC proposes a predictive action where the input pulse width parameter for the control action, T_{on} is given by Eq. 5 [14]

$$T_{on} = \frac{|D|}{K_{sa}} \tag{5}$$

For negative values of D, a rectangular input of negative amplitude can be applied to the system with the width of the input determined using the absolute value of D. Equation 5 can be used to calculate precisely the control action parameter in the absence of any system changes and/or sensor noise and disturbances. Assuming that the system $G(s)$ is unchanging, the presence of small system/sensor noise can result in the expected system steady state output Y_{ss} predicted by Eq. 5, to deviate from the expected value.

EMPC further improvises the control action in response to deviations by using the concept of iterative predictive action. In this algorithm, multiple rectangular inputs can be used to achieve zero steady state error. In each iteration, the rectangular input width is predicted using Eq. 5 for the new value of D, where demand D is difference between the current steady state value and the required original demand. The control action is applied to the system when the system is at steady state. Due to the use of multiple iterations, EMPC ensures zero steady state error.

2.1 Stability analysis

Let $G(s)$ be a system whose constant of proportionality has been estimated to be $K = K_{sa}$ using a single input-output mapping. This value of K is used for prediction of rectangular pulse width T_{on} using Eq. 5 to achieve the required steady state output determined by the demand D. Let us assume that the system $G(s)$ undergoes a change resulting in K changing from K_{sa} to $K_{sa}^{'}$. Since, EMPC uses Iterative Predictive Action for control without any adaptation mechanism for changes in the system, it continues to use $K = K_{sa}$ in Eq. 5 for prediction and control and this will result in errors.

Consider that the current system steady state output Y_{ss} is zero and the system is given a new reference value $R > 0$. Then the demand $D = R - Y_{ss} = R$ can be considered as the error e_0 at the end of the 0th iteration. Then,

$$e_0 = R \tag{6}$$

The width of the rectangular input is predicted using Eq. 5 as

$$T_{on1} = \frac{|e_0|}{K} \tag{7}$$

Since the current system proportionality constant is $K_{sa}^{'}$, the steady state output of the system for this input is given by

$$Y_1 = T_{on1} K_{sa}^{'} = e_0 \frac{K_{sa}^{'}}{K_{sa}} \tag{8}$$

Then the steady state error at the end of the first iteration is given as

$$e_1 = R - Y_1 = e_0 - Y_1 = e_0 \left(1 - \frac{K_{sa}^{'}}{K_{sa}}\right) \tag{9}$$

The iterative predictive action considers this error e_1 as the new demand in the next iteration and hence results in two main cases.

Case 1 When $K_{sa}^{'} < K_{sa}$, $e_1 > 0$ and $e_1 < e_0$. Iterative predictive action is used to calculate the width of the rectangular input to be fed in the second iteration to correct the error e_1. This value of T_{on2} is given by Eq. 10

$$T_{on2} = \frac{|e_1|}{K} \tag{10}$$

This results in

$$Y_2 = Y_1 + T_{on2} K_{sa}^{'} \tag{11}$$

Therefore, steady state error at the end of the second iteration is given by

$$e_2 = R - Y_2 = e_0 \left(1 - \frac{K_{sa}^{'}}{K_{sa}}\right)^2 \tag{12}$$

This results in $e_2 > 0$ and $e_2 < e_1 < e_0$. This can be repeated to obtain the steady state error at the end of the nth iteration as e_n where,

$$e_n = e_0 \left(1 - \frac{K_{sa}^{'}}{K_{sa}}\right)^n \tag{13}$$

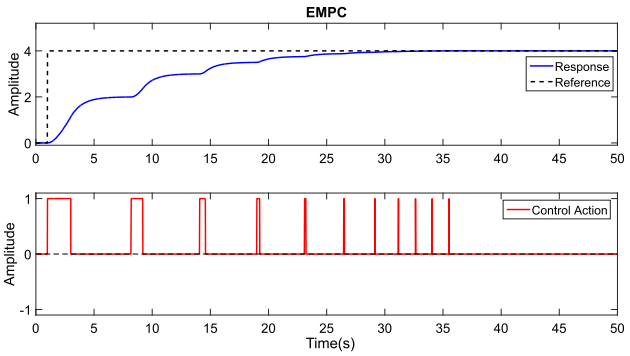

Fig. 2 Simulated response of $G(s)$ for $K_{sa}^{'} < K$

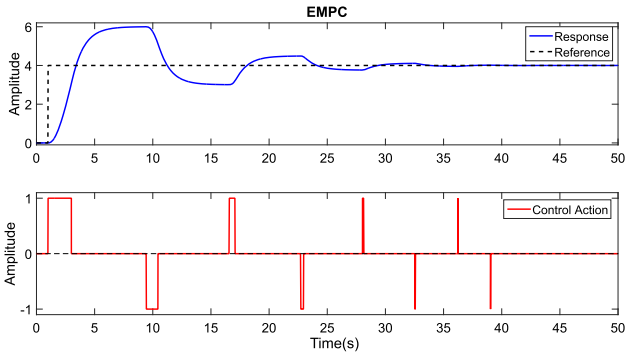

Fig. 3 Simulated response of $G(s)$ for $K < K_{sa}^{'} < 2K$

In this case, $e_n > \forall n \in N$ and $e_n < e_{n-1}$. The system response is similar to that of an over-damped system as $y(t)$ converges to the required steady state value with each iteration. Figure 2 shows typical response for $K_{sa}^{'} = 1$.

Case 2 When $K_{sa}^{'} > K$. This results in $e_1 < 0$. In the next iteration, e_1 is the new demand D and since $e_1 < 0$, a rectangular input of negative amplitude is applied to the system and the width of the input is predicted with Eq. 10 resulting in,

$$Y_2 = Y_1 - T_{on2} K_{sa}^{'} \tag{14}$$

The error at the end of the second iteration is given by Eq. 12, which is same as before. Therefore the nth iteration error for this case is also calculated from Eq. 13. However, since the sign of e_n alternates in Case 2, the magnitude of the steady state error at the end of each iteration is considered and given by Eq. 15

$$|e_n| = e_0 \left| 1 - \frac{K_{sa}^{'}}{K} \right|^n \tag{15}$$

The requirement to ensure stability with the change in the system parameters is that the magnitude of the error given by Eq. 15 should asymptotically go towards zero. This is possible when,

$$\left| 1 - \frac{K_{sa}^{'}}{K} \right|^n < 1 \Rightarrow -1 < \left(1 - \frac{K_{sa}^{'}}{K} \right)^n < 1 \tag{16}$$

Since $\frac{K_{sa}^{'}}{K} > 0$, it is implied that $(1 - \frac{K_{sa}^{'}}{K}) \not\geq 1$. Hence, $(1 - \frac{K_{sa}^{'}}{K}) > -1$ is the necessary condition. This can be simplified to obtain the stability criterion for EMPC for changes in system parameters in the absence of adaptation as shown in Eq. 17

$$K_{sa}^{'} < 2K \tag{17}$$

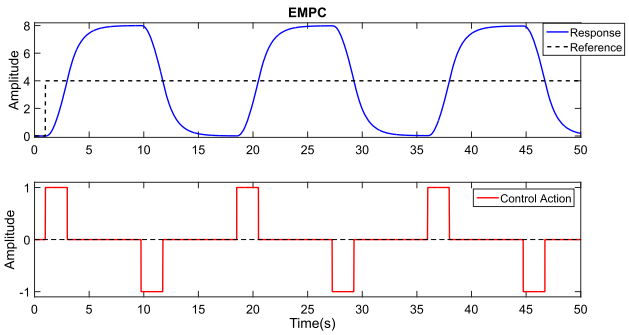

Fig. 4 Simulated response of $G(s)$ for $K_{sa}^{'} = 2K$

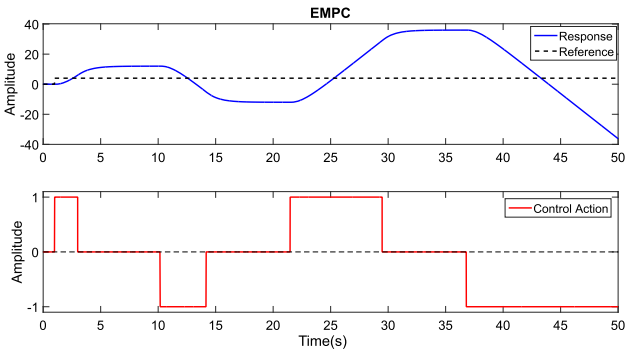

Fig. 5 Simulated response of $G(s)$ for $K_{sa}^{'} > 2K$

$K_{sa}^{'} < K$ results in case 1 whose response is seen in Fig. 2. This is a stable scenario with an over-damped response. $K_{sa}^{'} < 2K$ is another stable scenario with an under-damped response as seen in Fig. 3. Although, the system response oscillates around the required steady state value, the response slowly converges towards the required steady state value with each iteration. $K_{sa}^{'} = 2K$ results in the boundary condition for the stability of EMPC where the system response oscillates with the magnitude of the error at the end of each iteration remaining constant throughout as seen in Fig. 4. $K_{sa}^{'} > 2K$ results in instability as seen in Fig. 5.

The stability condition of Eq. 17 shows that EMPC is capable of controlling systems for a large range of K_{sa} values using the iterative predictive action control technique.

However, without any adaptation there is a deterioration in the performance as the value of $K_{sa}^{'}$ moves further away from K_{sa}, with multiple iterations and possible overshoots and oscillations being part of the response. Hence, it is necessary for EMPC to adapt to system parameter changes and improve the overall performance.

EMPC proposes an adaptation technique called on-job relearning (OJR) which improves upon the iterative predictive action to overcome these problems [15]. In this algorithm, at the end of an iteration, a ratio called parameter correction coefficient (PCC) is calculated as shown in Eq. 18

$$PCC = \frac{|D|}{Y_{ss} - Y_{ss0}} \tag{18}$$

where Y_{ss} is the current steady state output of the system after the predicted rectangular input is applied and Y_{ss0} is the steady state output value before the application of the rectangular input. The parameter correction coefficient can be used to adapt EMPC to correct the prediction appropriately for the next iteration. This is achieved by modifying Eq. 5 which is used to predict the width of the rectangular input to consist of the PCC term as given below

$$T_{on} = \frac{|D|}{K} PCC \tag{19}$$

The value of PCC is initially set to 1 and remains unchanged till EMPC needs to adapt to system changes. When the system parameters change and this results in a deviation between expected and obtained steady state response, PCC is calculated based on Eq. 18 accordingly. Due to this change in PCC in each iteration of control action, in some subsequent iteration of input, the system response will converge towards the demand. It is seen that when error finally reaches zero at some iteration, Eq. 18 will revert PCC back to 1. Therefore a more general form of PCC is proposed given by Eq. 20.

$$PCC_i = \frac{|D_i|}{Y_i - Y_{i-1}} PCC_{i-1} \tag{20}$$

Stability of OJR based adaptation using PCC can be evaluated by modifying the criteria for stability given by Eq. 17

When $K = K_{sa}^{'}$, the steady state value Y_1 is given by Eq. 8. PCC is calculated using Y_1 as

$$PCC_1 = \frac{|D|}{Y_1 - Y_0} PCC_0 \tag{21}$$

where $PCC_0 = 1$, $D = R$, $Y_0 = 0$. This simplifies as,

$$PCC_1 = \frac{R}{Y_1} = \frac{T_{on} K_{sa}}{T_{on} K_{sa}^{'}} = \frac{K_{sa}}{K_{sa}^{'}} \tag{22}$$

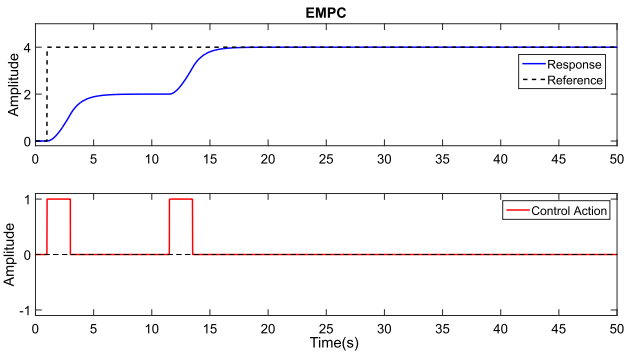

Fig. 6 Simulated response of $G(s)$ for $K < K_{sa}^{'} < 2K$

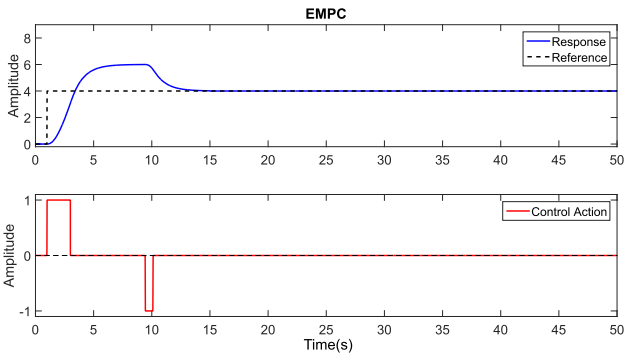

Fig. 7 Simulated response of $G(s)$ for $K < K_{sa}^{'} < 2K$

Equation 10 is now modified as,

$$T_{on2} = \frac{|e_1|}{K} PCC_1 = \frac{|e_1|}{K_{sa}^{'}} \tag{23}$$

The steady state output obtained at the end of the second iteration is given by Eq. 24, resulting in $e_2 = 0$.

$$Y_2 = Y_1 + T_{on2} K_{sa}^{'} = Y_1 + e_1 = R \tag{24}$$

Figures 6, 7, 8 and 9 show simulated response for EMPC with OJR. It can be seen that OJR is capable of adapting and controlling systems even when the parameter changes result in a violation of the earlier stability criterion of Eq. 17. Although, significant overshoot is present in the response of the system in these cases, the stability of the system is assured.

3 EMPC for the under-damped Type 1 system

An under-damped system has at least one complex conjugate pole pair. The general form of the under-damped Type 1 system is described by Eq. 25

$$H(s) = \frac{1}{s(as^2 + bs + c)} \times \frac{(s + z_1)(s + z_2) \cdots (s + z_m)}{(s + p_1)(s + p_2) \cdots (s + p_n)} \tag{25}$$

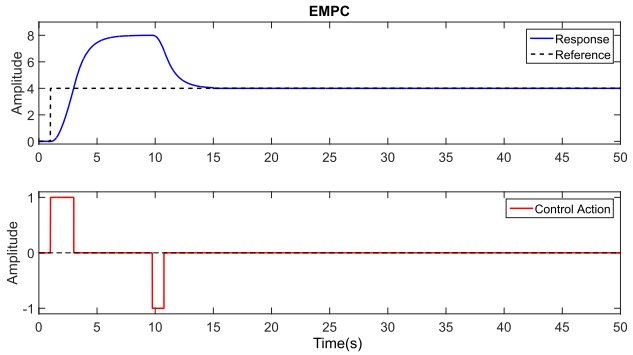

Fig. 8 Simulated response of $G(s)$ for $K_{sa}^{'} = 2K$

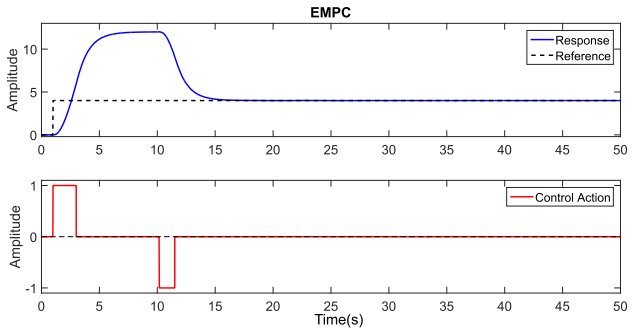

Fig. 9 Simulated response of $G(s)$ for $K_{sa}^{'} > 2K$

where $(z_i, p_j) \in \Re, i \in [1, m], j \in [1, n], m \leq (n+3)$ for a causal system, $b^2 < 4ac$ for existence of complex conjugate poles and $\frac{b}{2a} < p_j \ \forall \ j \in [1, n]$ for the conjugate poles to dominate the system response.

When a rectangular pulse is applied to an under-damped system, oscillations are caused due to the dominant complex conjugate poles. Hence EMPC proposed in the previous section is modified to reduce overshoots and oscillations.

EMPC therefore, proposes a control action $c(t)$ defined by Eq. 26, for the control of a Type 1 under-damped system [20]

$$c(t) = A_m \left[1 - \left\{\left(1 - e^{-\alpha(t-T_0)}\right) u(t - T_0)\right\}\right] u(t) \tag{26}$$

where A_m is the maximum allowed amplitude, α is a positive decay constant and T_0, analogous to T_{on} used for well-damped systems is the time-shift parameter. The Laplace Transform of $c(t)$ is given in Eq. 27

$$C(s) = \begin{cases} A_m \left[\frac{(1-e^{-T_0 s})}{s} + \frac{e^{-T_0 s}}{s+\alpha}\right], & T_0 \geq 0 \\ A_m \left(\frac{e^{\alpha T_0}}{s+\alpha}\right), & T_0 < 0 \end{cases} \tag{27}$$

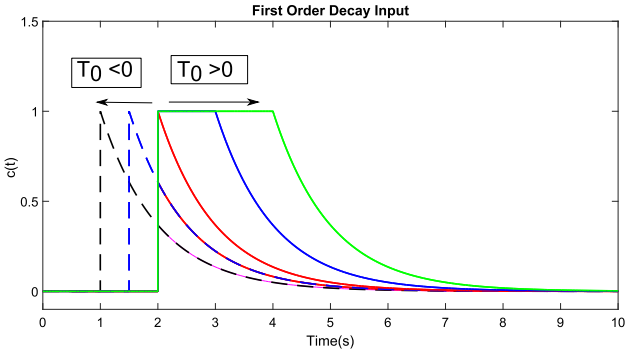

Fig. 10 Control input waveforms—first order decay signals shifted in time

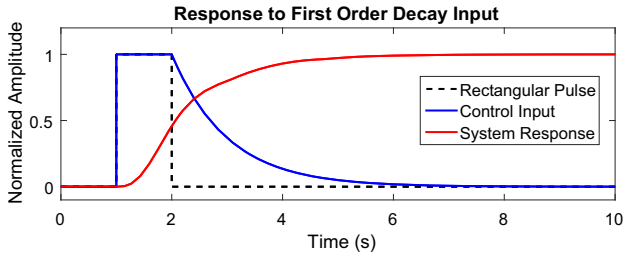

Fig. 11 Response of system H(s)=$\frac{17}{s(s^2+2s+17)}$ to first order decay input [$\alpha = 1$]

The output $Y(s)$ of a system $H(s)$, when the control input $C(s)$ is applied to it is given by Eq. 28

$$Y(s) = C(s)H(s) \tag{28}$$

Assuming zero initial conditions, the steady state value Y_{ss} can be obtained as in Eq. 29, using the final value theorem.

$$Y_{ss} = \begin{cases} A_m \frac{b_0}{a_0}(T_0 + \frac{1}{\alpha}) & T_0 \geq 0 \\ A_m \frac{b_0}{a_0} \frac{e^{\alpha T_0}}{\alpha} & T_0 < 0 \end{cases} \tag{29}$$

where, $b_0 = \prod_{i=1}^{m}(-z_i)$ and $a_0 = \left(\frac{c}{a}\right) \times \prod_{i=1}^{n}(-p_i)$

Similar to the well-damped system, for a given plant $H(s)$, a_0 and b_0 are constant. A_m is the maximum amplitude of input that can be applied to the system which is normalized for further analysis. Equation 29 can be simplified as,

$$Y_{ss} = \begin{cases} K_{sa}(T_0 + \frac{1}{\alpha}) & T_0 \geq 0 \\ K_{sa} \frac{e^{\alpha T_0}}{\alpha} & T_0 < 0 \end{cases} \tag{30}$$

Varying T_0 while keeping α constant results in the control input waveforms being shifted in time as shown in Fig. 10. The dotted lines represent the truncated portion of the exponential decay for $T_0 < 0$. The result of the control action on an under-damped system is shown in Fig. 11.

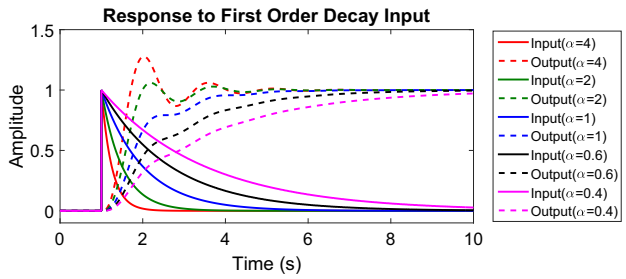

Fig. 12 Responses of system H(s)=$\frac{17}{s(s^2+2s+17)}$ to first order decay inputs [$T_0 = 0$] of various α

3.1 Choosing the decay constant α

In any system, the real part of the dominant pole determines the time constant of the system. The time constant of the under-damped system in Eq. 25 is $\frac{2a}{b}$. It can be seen that for $\alpha < \frac{b}{2a}$, the system shows no overshoot, as shown in Fig. 12. It may also be observed that the system rise time increases for smaller values of α [20,21]. For the learning phase, a value of α is chosen which satisfies the condition mentioned, and for which the system meets the rise time requirements.

3.2 Control action

For a given demand, the control action parameter T_0 is determined by modiyfing Eqs. 30–31

$$T_0 = \begin{cases} \frac{D}{K_{sa}} - \frac{1}{\alpha} & D \geq \frac{K_{sa}}{\alpha} \\ \frac{\ln \frac{\alpha D}{K_{sa}}}{\alpha} & D \leq \frac{K_{sa}}{\alpha} \end{cases} \tag{31}$$

α is fixed according to the conditions mentioned in Sect. 3.1. From Eq. 29, it is understood that for constant A_m, b_0, a_0 and α, the steady state position value Y_{ss} is dependent only on T_0.

It can be seen in Eq. 29, that for $T_0 \geq 0$, Y_{ss} has a linear relationship with T_0, but for $T_0 < 0$, Y_{ss} varies non-linearly with respect to T_0.

3.3 Stability analysis

For a given demand R, error e_0 is given by Eq. 15. The control action parameter T_{01} is calculated from Eq. 31. If the system proportionality constant changes to K'_{sa}, the steady state output of the system can be either of two cases as in Eq. 32, depending on the conditions of Eq. 31. Iterative predictive action is used here as well in cases of external disturbance or plant parameter changes.

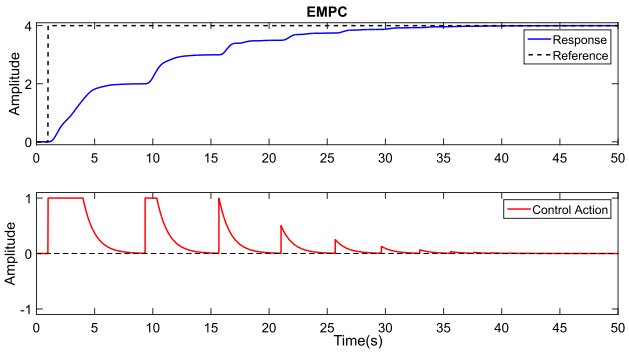

Fig. 13 Simulated response of $H(s)$ for $K < K'_{sa} < 2K$

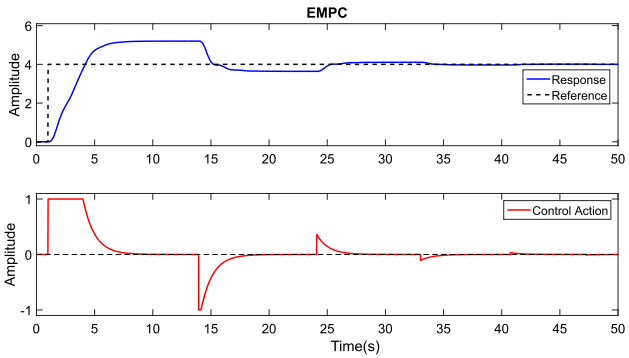

Fig. 14 Simulated response of $H(s)$ for $K < K'_{sa} < 2K$

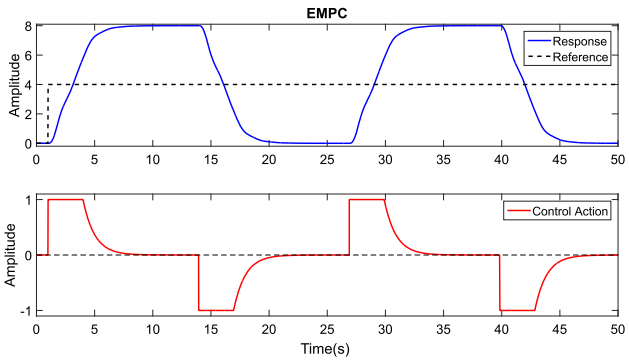

Fig. 15 Simulated response of $H(s)$ for $K'_{sa} = 2K$

$$Y_{ss} = \begin{cases} K'_{sa}(T_0 + \frac{1}{\alpha}) & T_0 \geq 0 \\ K'_{sa}\frac{e^{\alpha T_0}}{\alpha} & T_0 < 0 \end{cases} \tag{32}$$

Substituting Eq. 31 in Eq. 32 results in the earlier equation of Y_1 given by Eq. 8. Therefore, the stability criteria for the under-damped case will be the same as that of the well-damped case given by Eq. 17.

Figures 13, 14, 15 and 16 show simulated response for EMPC applied to the under-damped system $H(s)$ without adaptation.

Similar to the case of well-damped system, EMPC proposes OJR for the case of under-damped system as well by

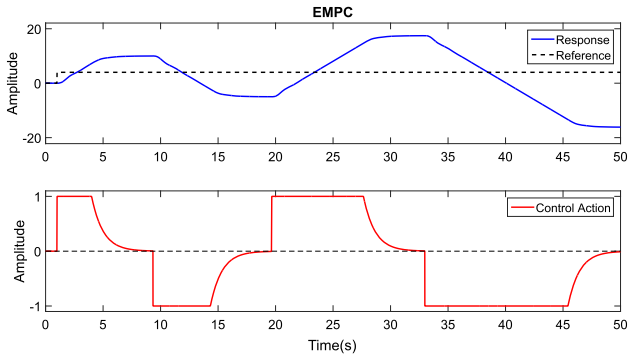

Fig. 16 Simulated response of $H(s)$ for $K_{sa}^{'} > 2K$

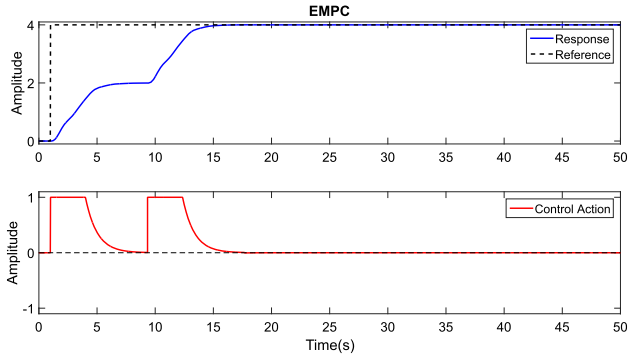

Fig. 17 Simulated response with OJR for $H(s)$ for $K < K_{sa}^{'} < 2K$

using the parameter correction co-efficient(PCC) shown in Eq. 33.

$$T_0 = \begin{cases} \frac{|D|}{K} PCC - \frac{1}{\alpha} & \frac{|D|}{K_{sa}} PCC \geq \frac{1}{\alpha} \\ \frac{\ln\left(\frac{\alpha|D|}{K_{sa}} PCC\right)}{\alpha} & \frac{|D|}{K_{sa}} PCC < \frac{1}{\alpha} \end{cases} \tag{33}$$

Figures 17, 18, 19 and 20 show simulated response for EMPC applied to the under-damped system $H(s) = \frac{17}{s(s^2+2s+17)}$ with adaptation. EMPC with OJR converges to zero steady state even in the case of $K > 2K_{sa}$ which surmounts the earlier set stability criterion and stability of the system is assured.

In both the well-damped and the under-damped case, EMPC with OJR results in a stable response to a given system. The results presented in this section have demonstrated that for change in system parameters EMPC with OJR is able to adapt. Since the adaptation occurs only in steady state, it is assumed that the system parameters do not change during the application of the control action. In the event of system parameter changes during the application of control action, EMPC with OJR will consider the average effect of these changes since it only records the final steady state value to calculate PCC. Hence EMPC might require more than 1 iteration to settle to zero steady state error.

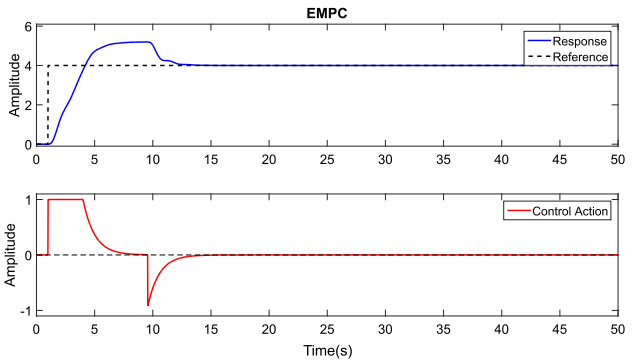

Fig. 18 Simulated response with OJR for $H(s)$ for $K < K_{sa}^{'} < 2K$

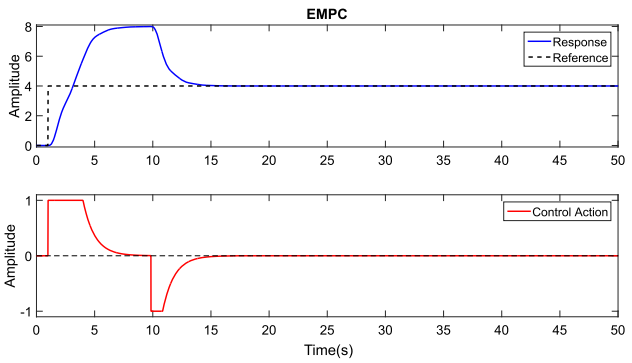

Fig. 19 Simulated response with OJR for $H(s)$ for $K_{sa}^{'} = 2K$

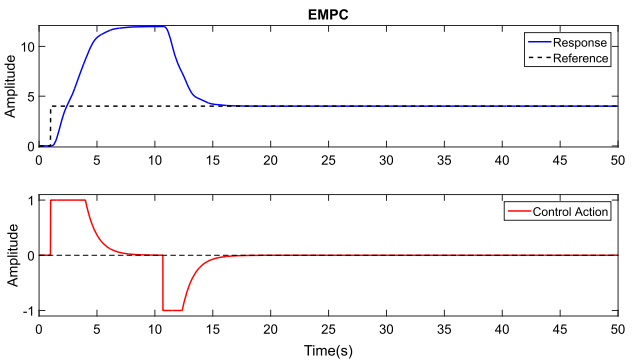

Fig. 20 Simulated response with OJR for $H(s)$ for $K_{sa}^{'} > 2K$

4 Comparision of EMPC and PD for stability

The PD controller is a closed loop error based classical controller that uses fixed gains on the error to give the control action. Typical PD transfer function is of the form,

$$P = \frac{K_p + sK_d}{s + \beta} \tag{34}$$

where K_p is the proportional constant and K_d is the differential constant. β is a far away pole to account for the causality of the controller and to attenuate high frequency noise from the feedback sensor.

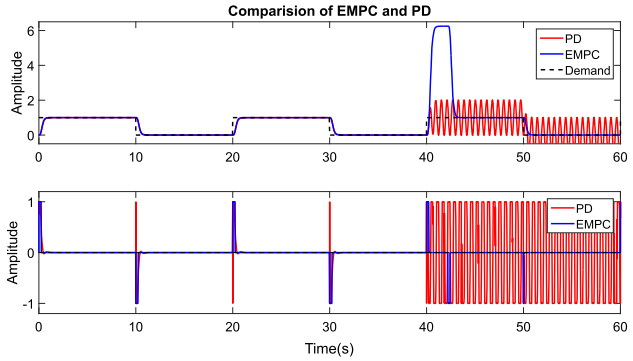

Fig. 21 Comparison of EMPC and PD for $G(s) = \frac{1000}{s(s^2+60s+500)(s^2+60s+500)}$

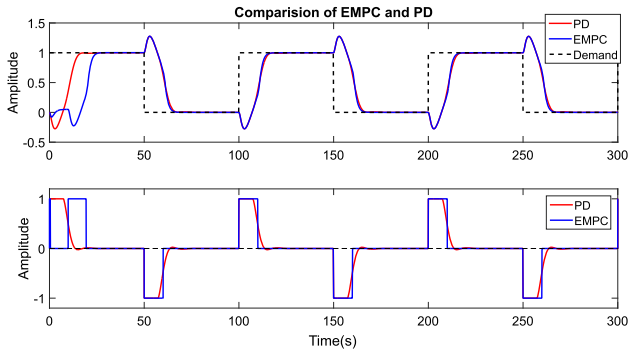

Fig. 22 Comparison of EMPC and PD for $G^{'}(s) = \frac{0.5(0.2-s)}{s(s^2+2s+1)}$

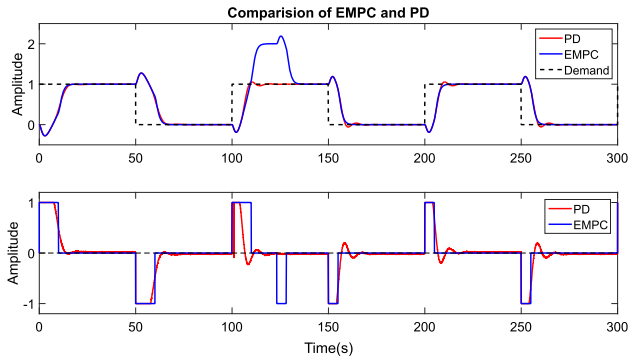

Fig. 23 Comparison of EMPC and PD for $G_1^{'}(s) = \frac{0.5(0.4-s)}{s(s^2+2s+1)}$

For the well damped system defined by $G(s)$, the poles are all on the real axis and on the left hand of the s-domain. The PD gains $K_p, K_d \in \Re$ can be chosen such that the system response is stable. But changes in a well-damped plant may result in the PD controller becoming unstable. Since PD is not an adaptive controller, it is not expected to perform optimally for a plant parameter change.

To illustrate the strength of the stability of the controller, a 4 pole system is considered for simulation as shown in Eq. 35

$$G(s) = \frac{1000}{s(s^2+60s+500)(s^2+60s+500)} \tag{35}$$

PD controller is critically tuned to get the best possible well damped response. The tuning is done to achieve least steady state error and minimum rise time for a given step input. and EMPC is also made to learn for this system and during the learning phase, the value of K_{sa} is determined to be 4. The first 40 seconds of Fig. 21 shows that both PD and EMPC perform optimally to give the best response. At the end of the 40th second, the system is changed to Eq. 36

$$G_1(s) = \frac{1000}{s(s^2+30s+200)(s^2+30s+200)} \tag{36}$$

which is a fairly big change. For this, PD fails to converge and goes into sustained oscillations as shown in Fig. 21. Due to OJR, EMPC is able to adapt by using PCC and determining the new $K_{sa}^{'}$ and is able to converge as shown in the figure. Thus EMPC with OJR is stable even in regions considered unstable for a traditional closed loop well damped stable system.

Consider the case of $G^{'}(s)$ to have a right hand plane zero as in the case of a non-minimum phase system, given by Eq. 37

$$G^{'}(s) = \frac{0.5(0.2-s)}{s(s^2+2s+1)} \tag{37}$$

Figure 22 shows comparison of system responses to PD and EMPC when the system contains a RHP zero. PD controller is tuned to get the best response for system $G^{'}(s)$. EMPC in the first iteration gives the input as learnt for the minimum phase system. At the end of the first iteration, the parameter correction co-efficient is calculated from Eq. 18. Adaptation due to OJR gives the correct input in the second iteration to reach steady error of zero. Figure 22 shows EMPC settles to the reference with zero steady state error in two iterations. For the next set of references, since EMPC has adapted well, it performs as good as a very well tuned PD in terms of rise time and initial overshoot. Figure 23 compares PD and EMPC for change in system parameter $K = 2K_{sa}$. It can be seen from the figure that initially both well tuned PD and EMPC with initial learning give the best comparable response. At $t = 100$ s, system parameter K is changed to twice its original value. Since PD is not an adaptive controller, it results in overshoots. On the other hand, EMPC in the first iteration overshoots since adaptation due to OJR occurs only at steady state. Once adapted, in the next and subsequent iterations, it is seen that EMPC controller results in optimal response for the changed system.

Consider an under-damped system defined by Eq. 38

$$H(s) = \frac{17}{s(s^2+2s+17)} \tag{38}$$

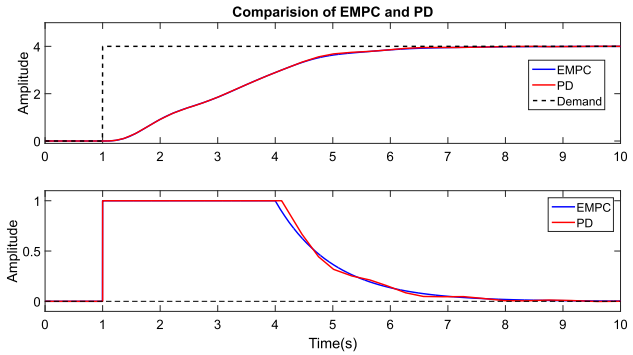

Fig. 24 Comparison of EMPC and PD for under-damped system $H(s) = \frac{17}{s(s^2+2s+17)}$

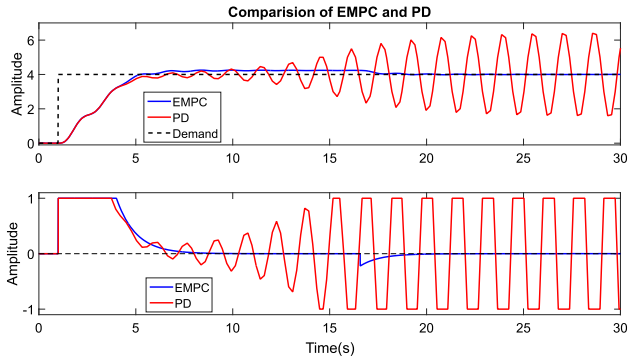

Fig. 25 Comparison of EMPC and PD for under-damped system $H^{'}(s) = \frac{17}{s(s^2+0.5s+16.0625)}$

consisting of at least two open-loop poles in the S domain and these two poles being dominant poles in the system response. From the root locus analysis, it is seen that the gain for the PD system has to be small < 2. Figure 24 shows the comparison of EMPC and PD for the system H(s). For higher gain, the system response will be unstable. Also, any change in the plant parameter will affect the system response significantly especially if the conjugate poles move more closer to the imaginary axis.

Figure 25 shows the comparison of system response for $H^{'}(s) = \frac{17}{s(s^2+0.5s+16.0625)}$ to EMPC and PD. It can be seen that PD is tending towards instability whereas EMPC using OJR has brought the system to zero steady state.

5 Control of a flexible shaft coupled to a DC motor based positioning system

A DC motor based positioning system with the load coupled through a flexible shaft, is a Type 1 system with complex conjugate poles. The mathematical model of this system is represented in Fig. 26. The typical transfer function for such a system is given by,

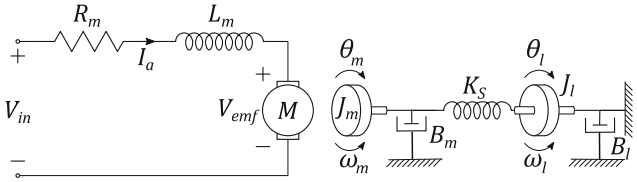

Fig. 26 Model of a DC motor position system coupled to a load with a flexible shaft

$$P(s) = \frac{K_s K_m}{s\big[(R_m + sL_m)(sJ_l + B_l)(s^2 J_m + sB_m + K_s) + K_s K_m^2 + sK_m^2(sJ_l + B_l) + K_s(R_m + sL_m)(sJ_m + B_m)\big]} \tag{39}$$

where K_s is spring constant of the shaft coupling, K_m is the motor torque constant, R_m is the armature resistance, L_m is the armature inductance, J_l is the load inertia, B_l is the viscous friction at the load, J_m is the motor inertia and B_m is the viscous friction at the motor shaft.

It can be seen from Eq. 39 that the system has five poles. The system, belonging to Type 1 family, has a pole at the origin. The electrical pole is located to the far left on the s-plane. The complex conjugate poles are contributed by the spring - inertia combination. The system is chosen such that the pole due to the motor mechanical time constant is faster in comparison to the complex conjugate poles, allowing the complex poles to dominate the system response. This system can be approximated to a second order Type 1 under-damped system.

5.1 Comparison of efficiency of different controllers

EMPC proposes two distinct control actions based on whether the system is well-damped or under-damped. Therefore we compare the two cases where the spring constant $K_s \gg 1000$ mNm/rad (well-damped) and $K_s << 100$ mNm/rad (under-damped)

5.1.1 Well-damped system

Table 1 is a typical well-damped setup. Since the spring constant is very high, there is stiff coupling between motor and load . Therefore Eq. 39 can be approximated to a Type-1 second order system defined by Eq. 40

$$P(s) = \frac{K_m}{s\big[(R_m + sL_m)(s(J_l + J_m) + (B_l + B_m))\big]} \tag{40}$$

For the specifications mentioned in Table 1, the poles of the system will lie at -1000, -25. Figure 27 compares the responses of a PD controller tuned for a critically damped response with that of EMPC. The tuning is done to achieve

Table 1 System specifications used for the Simulink model

Parameter	Value	Unit
R_m	2	Ω
L_m	2	mH
J_m	20×10^{-7}	Kgm^2
B_m	10^{-4}	Nm/(rad/s)
J_l	20×10^{-7}	Kgm^2
B_l	10^{-4}	rad/s
K_s	2000	mNm/rad
K_m	20	mNm/A

least steady state error and minimum rise time for a given step input. The input to the system shown in the subplot is normalised for easy representation. Substituting the specification values of the system into Eq. 40, it can be seen that $K_{sa} = 4$. Therefore EMPC calculates a control action pulse width of duration 0.25. From the system response it is seen that PD has a marginally lesser rise time than EMPC. The second subplot showing the control action input to the system clearly indicates that PD controller applies more input than EMPC to give the same response. To better understand the energy exchange, the third subplot shows the I^2R energy losses in the armature resistor due to application of control input. PD controller applies a brake by giving negative voltage input. this will result in more I^2R losses in the system. Hence EMPC is more efficient than PD.

In Fig. 27, the rise time of EMPC is more than that of PD. In [16], EMPC proposes another control action termed bipolar action. Here, during the learning phase, EMPC applies a pulse width of fixed duration followed by another pulse width in the negative direction until the motor stops moving. This effect of "braking" the system allows EMPC to give a longer pulse width to reach the demand faster than the unipolar case. The effective change for the given setup can be seen in Fig. 28. The second subplot of the figure shows the zoomed in view of the first subplot and it is seen that EMPC in bipolar mode has a rise time which is almost the same as that of critically damped response due to PD controller. Also, since EMPC now gives a negative voltage as well, the overall energy dissipation across the armature resistor has increased and almost the same as that of PD controller.

Figure 29 shows the comparison of PD and EMPC controller responses for a large inertia system where $J_l^{'} = J_l * 10$. Here, PD controller is retuned to get the critical damped response for the new system. Though the PD input changes significantly, the unipolar EMPC control action remains the same. This is because the system proportionality constant K_{sa} remains the same. Thus, with the same EMPC control action as before, the system is able to reach the required demand but at a much slower rate since the inertia dominates

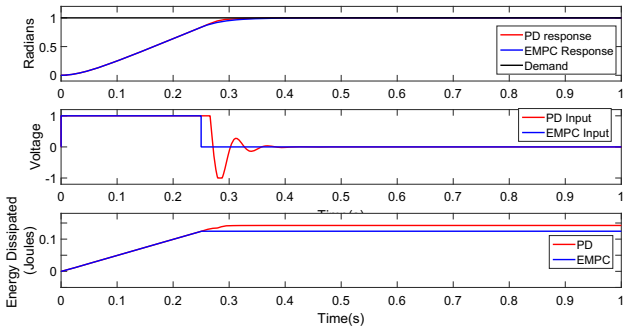

Fig. 27 Comparison of EMPC and PD for well-damped position control of motor

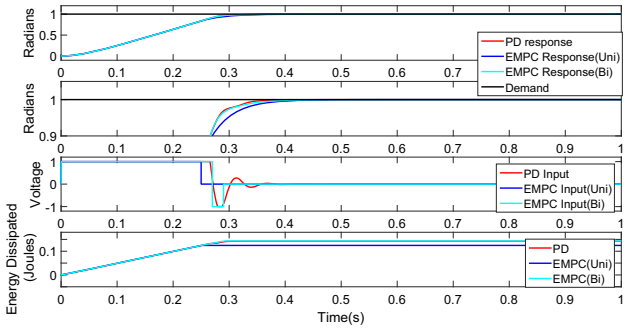

Fig. 28 Comparison of EMPC(bipolar) and PD for well-damped position control of motor

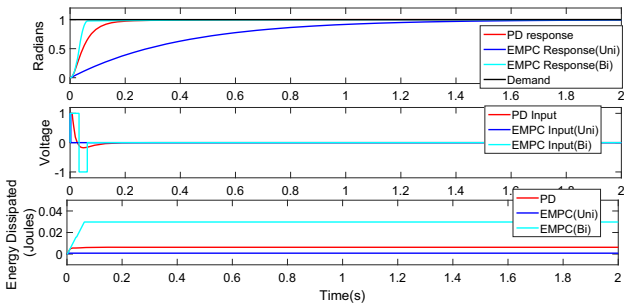

Fig. 29 Comparison of EMPC and PD for well-damped position control of motor for large inertia system $J_l^{'} = J_l * 10$

the response and the pole is now ten times nearer to the origin. On the other hand, bipolar mode of EMPC require relearning. Figure 29 shows that bipolar mode response is faster than PD since EMPC now has a larger control over the braking period. As expected, the energy losses in the armature resistance now reflect these input changes. EMPC in unipolar is the most efficient use of input energy though it has a larger rise time. PD comes second and EMPC in bipolar mode dissipates largest energy due to the initial high pumping of energy into the inertia and later applying a large brake to get the best rise time.

Figure 30 shows comparison of system response of EMPC and MRAC controllers applied to the same system whose specifications are mentioned in Table 1 both of which have

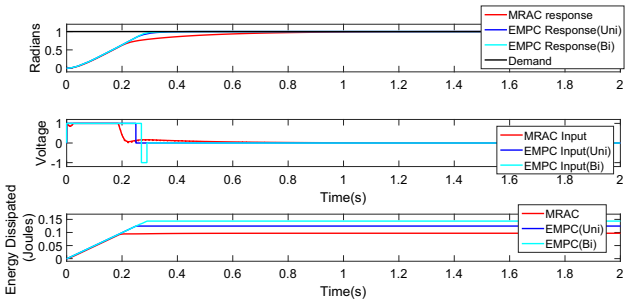

Fig. 30 Comparison of EMPC and MRAC for well-damped position control of motor

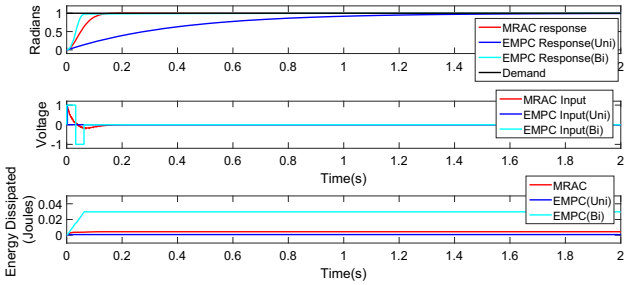

Fig. 31 Comparison of EMPC and MRAC for well-damped position control of motor for large inertia system $J_l^{'} = J_l * 10$

been previously put through their respective learning phases and adapted to get the best response. The MRAC model and tuning were presented earlier in [14,15] and the same has been used here for comparison with EMPC. It can be seen that system response due to MRAC has a higher rise time than EMPC. Also, the energy losses in the armature resistor of EMPC and MRAC are in a comparable range. The higher dissipation by EMPC controller can be attributed to more energy input provided to the system to improve the rise time.

Figure 31 compares the system response of EMPC and MRAC controllers applied to a modified system where $J_l^{'} = J_l * 10$. MRAC now has better rise time compared to unipolar EMPC control but still lesser than EMPC in bipolar mode. The energy dissipated by MRAC controller in the armature resistor is greater than EMPC in unipolar mode but lesser than EMPC in bipolar mode. EMPC therefore provides a way to compromise on either a good rise time by using Bipolar control action or a energy efficient input by using an Unipolar control action.

5.1.2 Under-damped system

For the under-damped system we consider Eq. 39. The various parameters are as per Table 2. EMPC for the under-damped system proposes a first order decaying input to be the control action to the system [20]. The parameters for control have been discussed in [20] and for the system simulated here, the parameters are learnt based on previous literature.

Table 2 Under damped system specifications used for the Simulink model

Parameter	Value	Unit
R_m	3	Ω
L_m	3	mH
J_m	1×10^{-7}	Kgm^2
B_m	10^{-3}	Nm/(rad/s)
J_l	200×10^{-7}	Kgm^2
B_l	10^{-4}	rad/s
K_s	20	mNm/rad
K_m	20	mNm/A

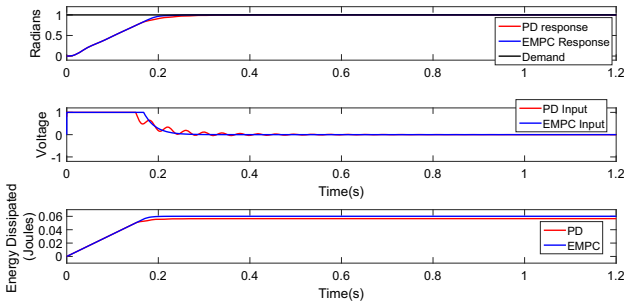

Fig. 32 Comparison of EMPC and PD for under-damped position control of motor

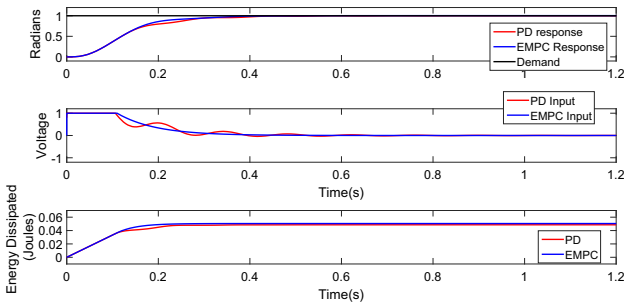

Fig. 33 Comparison of EMPC and PD for under-damped position control of motor for large inertia system $J_l^{'} = J_l * 10$

It is seen from Figs. 32 and 33, that tuning PD for an open loop under-damped system is difficult to get a critical damping in the system response. Especially the final tuning is very sensitive to change in system parameters and can lead to unstable outputs if not tuned well. Figures shows that EMPC for under-damped system gives a smooth response unlike that of PD. Though the subplots indicating the energy dissipated in the armature resistors show that the efficiency is the same, PD controller can cause more transient noise and high frequency noises maybe introduced into the driver circuit unlike EMPC which gives a smooth decaying input. Hence EMPC would give a better motor driver performance compared to that of PD controller.

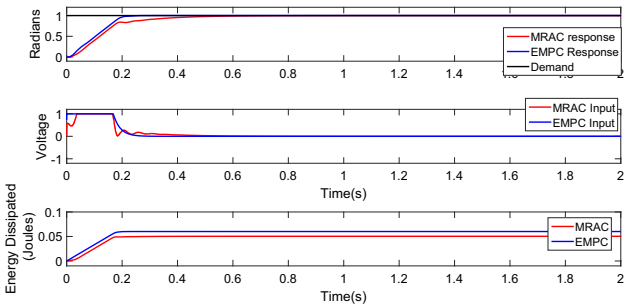

Fig. 34 Comparison of EMPC and MRAC for under-damped position control of motor

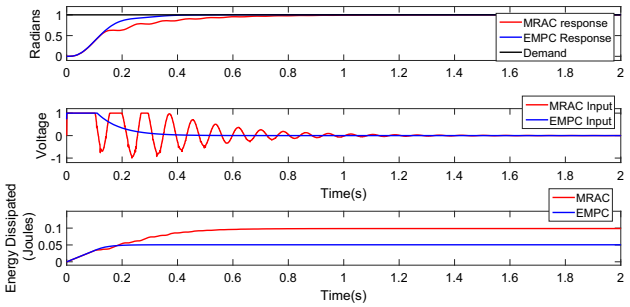

Fig. 35 Comparison of EMPC and MRAC for under-damped position control of motor for large inertia system $J_l^{'} = J_l * 10$

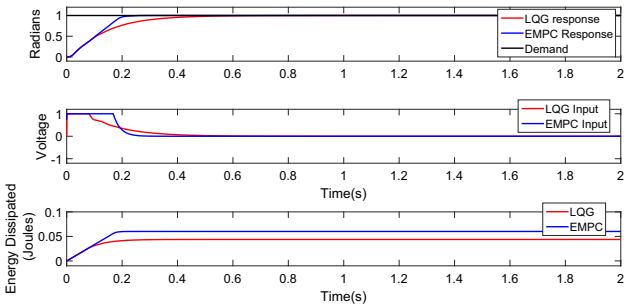

Fig. 36 Comparison of EMPC and LQG for under-damped position control of motor

Figures 34 and 35 compare the MRAC and EMPC for the same under-damped system. When the load inertia is very low as mentioned in Table 2, MRAC gives a good control action which results in a steady state response which has a smooth rise shown in Fig. 34. But the rise time of the system due to MRAC is higher than that of EMPC. Since MRAC does not give any sudden changes in input voltage, it performs almost as efficiently as EMPC for this system.

In Fig. 35, it can be seen that MRAC is unable to give an optimal input to the system even after sufficient iterations of learning. Further, the input given is chopping in nature and causes heavy losses in both armature resistance as well as the inductive losses in the coil.

Figures 36 and 37 compares linear quadratic Gaussian(LQG) controller with EMPC. The LQG controller

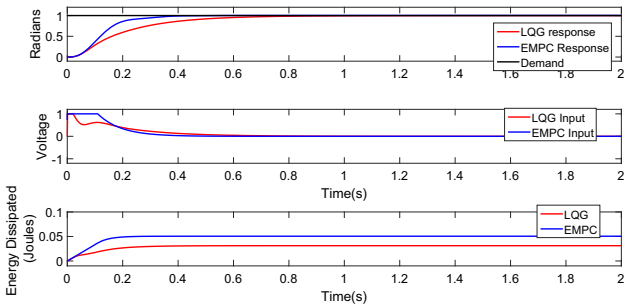

Fig. 37 Comparison of EMPC and LQG for under-damped position control of motor for large inertia system $J_l^{'} = J_l * 10$

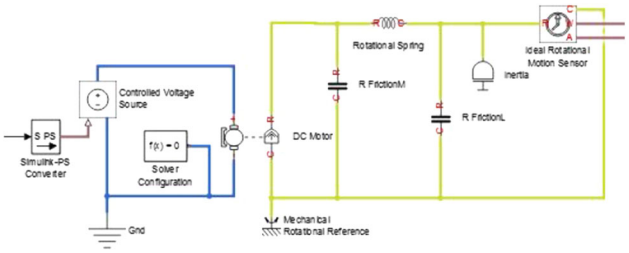

Fig. 38 Simulation model of a load coupled to a motor through a flexible shaft

consists of a linear quadratic regulator(LQR) controller along with a state estimator to predict required states from one measured state. The LQR controller was constructed based on the model from [15] and motor values changed to reflect the specifications in Table 1.The matrices Q and R were tuned manually to get the best possible response. EMPC has better performance than LQG in terms of the rise time. The energy dissipated also matches closely with that of LQG.

6 Control of a non-linear system DC motor based positioning system

Figure 38 shows a simulation model on SIMULINK of practical non-linear setup consisting of a load coupled to a motor through a flexible shaft. The model consists of a rotational spring placed in between the motor and an inertial element. Rotational elements *RFrictionM* and *RFrictionL* consisting of a dry friction component and a viscous friction component are placed on either side of the rotational spring to create an under-damped response. The values chosen for the simulation model are shown in Table 2. The additional frictions added have values shown in Table 3.

6.1 Design of EMPC

For practical systems which are based on the transfer function model shown in Eq. 39, [15,16,20] show that system constant of proportionality K_{sa} varies with demand and con-

Table 3 Friction specifications used for the Simulink model

Parameter	Value	Unit
$RFrictionM - Dry$	10	mNm
$RFrictionL - Dry$	2	mNm
$RFrictionM - Viscous$	$1e{-}03$	mNm
$RFrictionL - Viscous$	$1e{-}04$	mNm

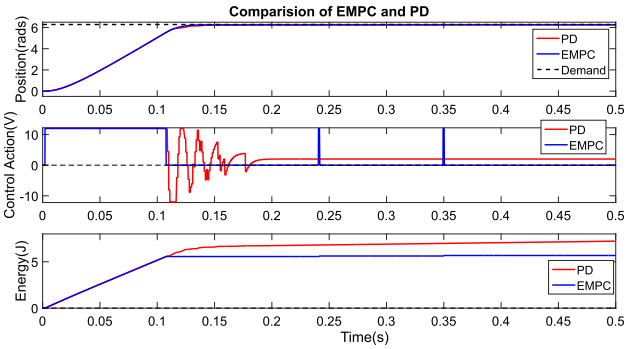

Fig. 39 Comparison of EMPC and PD for well-damped position control of a practical motor system

trol action parameter T_0. The non-constant value of K_{sa} can be attributed to the non-linearities present in practical systems like dry friction and stiction.

Therefore, EMPC proposes the use of an Experience Mapped Knowledge(EMK) which is a one to one mapping of the control action to the final steady state value achieved due to the corresponding control action. In the learning phase, the EMK is populated by applying different input values of T_0 to the system and recording the final steady state value reach by the system. After learning is completed, during the application of the control action when a demand is given, EMPC will refer to the EMK and interpolate the required value of T_0 for the given demand. This method of using an EMK for a practical system has been shown to be robust for different demands and also shown to adapt to changes in system parameter [15,16,20].

6.2 Comparison of efficiency of different controllers

6.2.1 Well-damped system

For the well-damped case, the spring constant value of the rotation spring was made 3000 mNm/rad to make it stiff.

Figure 39 shows the comparison of a critically tuned PD controller with EMPC. Like in the ideal case presented in Sect. 5, performance in terms of rise time and settling time of EMPC matches that of PD. But due to the now introduced non-linearities, the energy dissipated in the motor is higher in the case of PD due to the effect of Derivative part of the controller. The third subplot shows the difference in energy

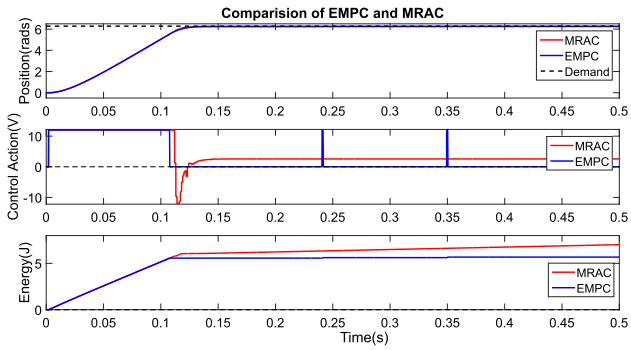

Fig. 40 Comparison of EMPC and MRAC for well-damped position control of a practical motor system

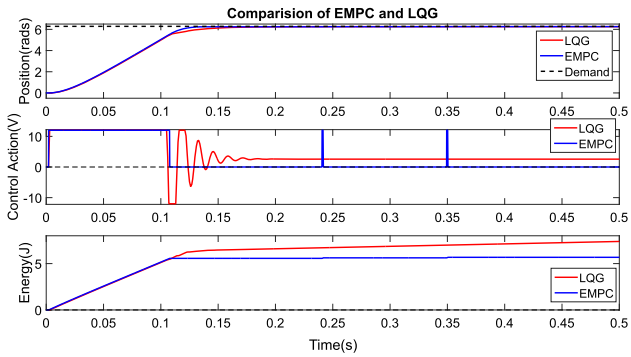

Fig. 41 Comparison of EMPC and LQG for well-damped position control of a practical motor system

dissipated to be about 20% that of EMPC. Therfore, EMPC is shown to give the best possible input to have the least energy loss.

Figure 40 shows the comparison of MRAC controller with EMPC. The reference signal to MRAC was determined intuitively to give the fastest possible rise time and a low value of learning rate ($\gamma < 0.01$) was used to let the controller adapt itself to give the best possible input for a given demand. Similar to that of PD, EMPC matches the performance of MRAC. Unlike the Derivative part of PD controller, MRAC continously adapts with a fixed learning rate and hence does not cause a chopping effect in the input. Therefore, the energy dissipation is lesser but still significantly higher than EMPC (about 10%) due to negative input given to brake the system.

Figure 41 shows the comparison of linear quadratic Gaussian(LQG) controller with EMPC. From the results, it is seen that EMPC also closely matches the response of LQG controller. LQG controller like MRAC also applies a negative voltage to bring the motor system to a halt and hence dissipates more energy. Thus EMPC proves to be better in terms of energy usage.

It should also be noted that in the case of all these controllers, the input is not terminated once steady state is reached. This is due to the presence of static(dry) friction which cannot be overcome with a small input. On the other

hand, EMPC terminates the input once a pulse width of T_0 as calculated from the EMK is applied. Though in the case of the other controllers, a simple threshold based switch can be introduced to terminate the input, in its native form, the controllers tend to waste a lot of energy due to their control action. In practical systems, this would lead to highly in-efficient usage of power. This issues is clearly resolved in EMPC in its basic algorithm where the input is terminated when the steady state error is within a suitable threshold.

6.2.2 Under-damped system

For the under-damped case, the spring constant value of the rotation spring was made 20 mNm/rad to make it flexible. Figure 42 shows the comparison of a PD controller with EMPC. Due to the non-linearities, it is very difficult to tune a PD controller for an under-damped system. The tuning is done to achieve least steady state error, minimum overshoots and minimum rise time for a given step input. The response from a PD controller tends to be marginally stable in most cases. The energy dissipated in the motor therefore is extremely high in the case of PD. The difference in energy dissipated to be more than twice that of EMPC. This will further become worse for PD controller if system parameters like load inertia is changed during operation.

Figure 43 shows the comparison of MRAC controller with EMPC. The reference signal to MRAC was chosen to have a slower rise time since MRAC is preferred to be used only on a well-damped linear system. A very low value of learning rate ($\gamma < 0.0001$) was used to let the controller adapt itself to give the best possible input for a given demand. MRAC performs better than PD controller due to its learning capability. But due to the conjugate poles of the under-damped system being triggered, MRAC tends to have spikes in its input caused by oscillations in the speed. Therefore, the energy dissipation is significantly higher than EMPC (about 20%) due to spikes in the input.

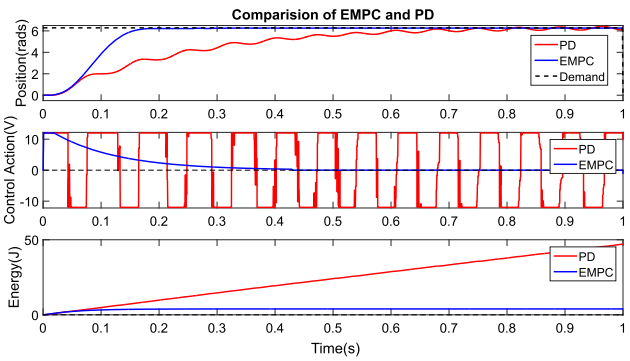

Fig. 42 Comparison of EMPC and PD for under-damped position control of a practical motor system

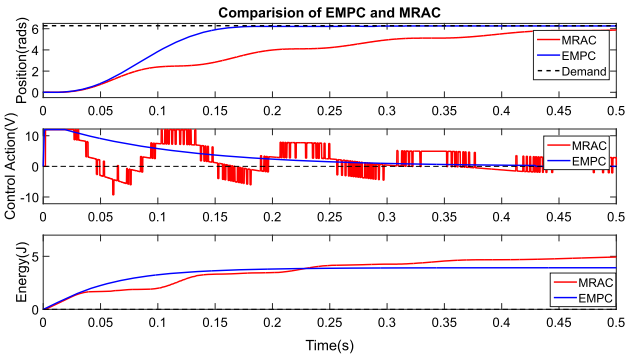

Fig. 43 Comparison of EMPC and MRAC for under-damped position control of a practical motor system

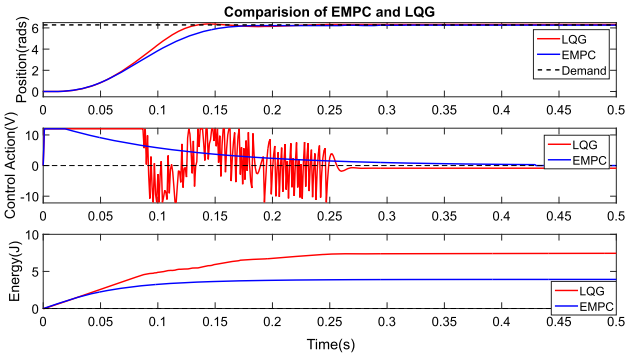

Fig. 44 Comparison of EMPC and LQG for under-damped position control of a practical motor system

Figure 44 shows the comparison of LQG controller with EMPC. From the results, it is seen that LQG controller performs much better than MRAC and PD. In the case of the under-damped system, LQG controller ends up applying an agressive control action causing spikes in the input to the motor. EMPC on the other hand provides a smooth control action and hence proves to be better in terms of energy usage.

In the case of the under-damped system, another key area of efficiency comes from reducing switching losses. It is clearly seen that MRAC,PD and LQG controllers input a high switching control action. EMPC gives a very smooth input and therefore has a very low switching losses. This further increases the life of the motor and driver system in a practical setup.

7 Computational cost of controllers

Modern controllers operate in the digital domain to benefit by the advancements in micro-controller technology. Computational cost depends on the number of mathematical operations to be performed and hence the overall computation time taken to implement a control algorithm becomes very important in effective control of a system. Based on the

system time constant, the sampling duration is fixed and the entire control algorithm should be executed within one sample duration for proper control action. Therefore selection of a micro-controller for a given algorithm becomes critical, which is decided based on the number of floating operations, multiplications, divisions, matrix operations required for the given algorithm. Higher the computation requirements, higher will be the cost of the controllers to be used. Following paragraphs try to assess the computational requirements of various control algorithms considered in the paper.

Till recently, the most widely used controller in the digital domain is the PID controller due to its ease of implementation. The PID controller requires just about 3 multiplications (One each for Proportional, Derivative, Integral). A final summation gives the control action to be implemented. In the category of adaptive controller, MRAC ,LQG and EMPC can be compared to PID in terms of computations.

In the case of MRAC and LQG, both use matrix multiplications.The order of the matrix is depends on the number of poles and zeros in the system. For a higher order system, there maybe a requirement of a Digital Signal Processor(DSP) to compute inverses of these matrices. For a typical 2nd order Type-1 system like that of a position control of DC motor, though a DSP is not required, the calculation of positive definite matrices for MRAC or the LQR gain in LQG still require more than 20–30 multiplications. Also, the time taken for computing inverse of a matrix is fairly large compared to just multiplication of two matrices. There is also a requirement to store intermediate results in the RAM of the controller and this has to happen at a fast rate.

In EMPC, the computational process can be seen majorly in two phases.

The learning phase involves application of input with one parameter varying (T_0) in incremental steps. The system waits for steady state output. The system is actually idle until this point and will consume the least power. Once steady state is reached, the value is directly stored in the ROM (such as EEPROM,FLASH) of the micro-controller along with the input parameter value for that iteration. The final length of this EMK stored in the memory depends on the resolution and operational range of the system. Saikumar and Dinesh [14] explains the mapping of position and input parameter T_0 to be linear for a major region and hence the points stored in the EMK for this region can be far apart and less in number. The required value during application for given demand is interpolated between the points in the EMK that it lies. The interpolation is a simple operation of taking the slope and calculating the required input parameter to be given to the system. A typical memory location required for storing one iteration of input in the learning phase would be 2 bytes each for the input and output. A memory of 1Kb would give atleast 500 points. A typical general purpose micro-controller usually ships with an inbuilt user memory of 1MB which is far more than necessary for EMPC.

In the application phase, for the given demand, EMPC interpolates from the EMK with two math operations consisting of multiplication and division and applies the control action and waits for the system to settle. In the case of OJR, at the end of an iteration, another multiplication is performed to get PCC which is applied in subsequent iterations. In the case of the under-damped system, a third multiplication occurs during the application of control action to calculate the first order decay input [20].

Thus EMPC requires as much computational power as that of a PD controller and a small memory unit to house the EMK. A simple 16 bit general purpose micro-controller capable of doing basic math operations is sufficient specification for a EMPC based controller.

8 Conclusion

A stability criterion for experience mapping based predictive controller (EMPC) applied to a Type 1 system was derived. The stability criterion developed was tested for both the well damped Type 1 systems and under-damped Type 1 systems. EMPC with OJR was shown to assure stability beyond this criteria. The simulation results of EMPC for DC motor based positioning system with a load coupled through a flexible shaft are presented as a case study prove stability even for system parameter changes. The efficiency of EMPC on a practical system was analysed in terms of energy dissipated in the armature resistance of the motor. In practical cases, EMPC was proven to be the best among PD,MRAC and State Space based controllers like the LQR and LQG controllesr. EMPC was also proven to require least computational power and memory requirements compared to other controllers, especially those that use matrix based equations.

References

1. Wu G-Q, Wu S-N, Bai Y-G, Liu L (2013) Experimental studies on model reference adaptive control with integral action employing a rotary encoder and tachometer sensors. Sensors 13:4742–4759
2. Bature AA, Muhammad M, Abdullahi AM (2013) Position control of a DC motor: an experimental comparative assesment between fuzzy and state feedback controller. ARPN J Eng Appl Sci 8:984–987
3. Sahin M, Bulbul H, Colak I (2012) Position control of a DC motor used in solar panels with artificial neural network. In: 2012 11th international conference on machine learning and applications (ICMLA), vol 2, pp. 487-492
4. Sastry S, Bodson M (2011) Adaptive control: stability, convergence and robustness. Courier Corporation
5. Shin H-C, Choi S-B (2011) Position control of a two-link flexible manipulator featuring piezoelectric actuators and sensors. Mechatronics 11:707–729

6. Liu YE, Skormin VA, Liu Z (2002) Experimental comparison on adaptive control schemes. In: Proceedings of the 27th IEEE conference on decision and control
7. Ruderman M, Krettek J, Hoffmann F, Bertram T (2008) Optimal state space control of dc motor. In: Proceedings of the 17th world congress the international federation of automatic control, pp 5796–5801
8. Yu G-R, Hwang R-C (2004) Optimal PID speed control of brushless DC motors using LQR approach, systems, man and cybernetics. In: 2004 IEEE international conference, vol 1, pp 473–478
9. Syahidah W, Rosli O, Joraimee MA, Norhidayah A (2014) Linear quadratic gaussian (LQG) Controller design for servo motor. Aust J Basic Appl Sci 8:700–713
10. Iwasaki T, Kataoka T (1989) Application of an extended kalman filter to parameter identification of an induction motor. In: Industry applications society annual meeting, 1989., conference record of the 1989 IEEE, pp 248–253
11. Aravind MA, Niranjan S, Dinesh NS, (2017) Optimal position control of a DC motor using LQG with EKF. In: 2017 international conference on mechanical, system and control engineering (ICMSC)
12. Chen CL, Feng G, Guan XP (2005) Delay-dependent stability analysis and controller synthesis for discrete-time fuzzy systems with time delays. IEEE Trans Fuzzy Syst 13(5):630–643
13. Han Q-L (2009) Improved stability criteria and controller design for linear neutral systems. Automatica 45:1948–1952
14. Saikumar N, Dinesh NS, (2012) Position control of DC motors with experience mapping based prediction controller. In: IECON, pp 2394 - 2399
15. Saikumar N, Dinesh NS, (2012) A study of experience mapping based prediction controller for position control of DC motors with inertial and friction load changes. In: 7th IEEE international conference on industrial and information systems
16. Saikumar N, Dinesh NS (2014) A study of bipolar control action with EMPC for the position control of DC motors. Int J Dyn Control 4:154–166
17. Saikumar N, Dinesh NS, Kammardi P (2015) Experience mapping based prediction controller for the smooth trajectory tracking of DC motors. Int J Dyn Control 5:704–720
18. Raghu CV, Dinesh NS, (2017) DC motor speed control using experience mapping based prediction controller (EMPC). In: 3rd international conference on control, automation and robotics (ICCAR)
19. Raghu CV, Dinesh NS (2017) EMPC for DC motor based tracking applications. In: IEEE 3rd international symposium in robotics and manufacturing automation (ROMA)
20. Aravind MA, Dinesh NS, Rajanna K (2017) Application of EMPC for under-damped type-1 systems. In: 3rd international conference on control, automation and robotics (ICCAR)
21. Aravind MA, Dinesh NS, Rajanna K (2018) Adaptive experience mapping based predictive controller for under-damped type 1 systems. Int J Dyn Control, pp 1–18, Springer, Berlin
22. Flanagan JR, Vetter P, Johansson RS, Wolpert DM (2003) Prediction precedes control in motor learning. Curr Biol 13:146–150
23. Ungerleider L, Doyon J, Karni A (2002) Imaging brain plasticity during motor skill learning. Neurobiol Learn Mem 78:553–564
24. Brashers-Krug T, Shadmehr R, Bizzi E (1996) Consolidation in human motor memory. Nature 382:252255
25. Shadmehr R, Holcomb HH (1997) Neural correlates of motor memory consolidation. Science 277:821825