Soft Comput (2009) 13:23–29
DOI 10.1007/s00500-008-0289-9

ORIGINAL PAPER

ADALINE based robust control in robotics: a Riemann-Liouville fractional differintegration based learning scheme

Mehmet Önder Efe

Published online: 13 March 2008

Abstract This paper presents an approach to improve the performance of intelligent sliding model control achieved by the use of a fundamental constituent of soft computing, named Adaptive Linear Element (ADALINE). The proposed scheme is based on the fractional calculus. A previously considered tuning scheme is revised according to the rules of fractional order differintegration. After a comparison with the integer order counterpart, it is seen that the control system with the proposed adaptation scheme provides (1) better tracking performance, (2) suppression of undesired drifts in parameter evolution and (3) a very high degree of robustness and insensitivity to disturbances. The claims are justified through some simulations utilizing the dynamic model of a two degrees of freedom (DOF) direct drive robot arm and overall, the contribution of the paper is to introduce the fractional order calculus into a robust and nonlinear control problem with some outperforming features that are absent when the integer order differintegration operators are adopted.

Keywords Fractional adaptation laws · Adaptive sliding mode control

1 Introduction

The measure of similarity for determining the activation degree of a neuron in a neural network structure or the crisp decision in a fuzzy system are all performed by Adaptive Linear Elements (ADALINE), which are the building blocks of computationally intelligent systems, as discussed in detail by Haykin (1994) and Jang et al. (1997). The research on these structures dates back to the McCulloch-Pitts neuron in 1940s. Although the structure was used in a wide scope of applications in different names, the essence of the problem has been the design of the learning by ADALINEs. Given a task to be accomplished, the process describing the best evolution of the adjustable parameters is the process of learning, which is sometimes called adaptation, tuning, adjustment or optimization, all referring to the same reality in the context of soft computing. Many approaches have been proposed, perceptron learning, gradient descent, Levenberg-Marquardt technique, Lyapunov based techniques are just to name a few, a good treatment can be found in Jang et al. (1997). A common feature of all these methods is the fact that the differentiation and integration, or shortly differintegration, of quantities are performed in integer order, i.e. $\mathbf{D} := \frac{\mathrm{d}}{\mathrm{d}t}$ for the differentiation with respect to t and $\mathbf{I} = \mathbf{D}^{-1}$ for integration over t in the usual sense. A significantly different branch of mathematics, called fractional calculus, suggests operators $\mathbf{D}^{\beta}$ with $\beta \in \Re$, (Oldham and Spanier 1974; Podlubny 1998), and it becomes possible to write $\mathbf{D}f = \mathbf{D}^{\frac{1}{2}}(\mathbf{D}^{\frac{1}{2}}f)$. Expectedly, Laplace and Fourier transforms in fractional calculus are available to exploit in closed loop control system design, involved with s^{β} or $(j\omega)^{\beta}$ generic terms, respectively (Das 2008).

Fractional calculus and dynamics described by Fractional Differential Equations (FDEs) are becoming more and more popular as the underlying facts about the differentiation and integration is significantly different from the integer order counterparts and beyond this, many real life systems are described better by FDEs, e.g. heat equation, telegraph equation and a lossy electric transmission line are all involved with fractional order differintegration operators. A majority of works published so far has concentrated on the fractional variants of the PID controller, which has fractional order differentiation and fractional order integration, implemented for

M. Ö. Efe (✉)
Department of Electrical and Electronics Engineering,
TOBB Economics and Technology University, Sögütözü Cad.
No 43, 06560 Sögütözü, Ankara, Turkey
e-mail: onderefe@etu.edu.tr

the control of linear dynamic systems, for which the issues of parameter selection, tuning, stability and performance are rather mature concepts utilizing the results from complex analysis and frequency domain methods of control theory (See Matignon 1998) than those involving the nonlinear models (See Momani and Hadid 2004) and parameter changes in the approaches.

Parameter tuning in adaptive control systems is a central part of the overall mechanism alleviating the difficulties associated with the changes in the parameters that influence the closed loop performance. Many remarkable studies are reported in the past and the field of adaptation has become a blend of techniques of dynamical systems theory, optimization, heuristics (intelligence) and soft computing. Today, the advent of very high speed computers and networked computing facilities, even within microprocessor based systems, tuning of system parameters based upon some set of observations and decisions has greatly been facilitated. In Åström and Wittenmark (1995), an in depth discussion for parameter tuning in continuous and discrete time is presented. Particularly for gradient descent rule for model reference adaptive control, which is considered in the integer order in Åström and Wittenmark (1995), has been implemented in fractional order by Vinagre et al. (2002), where the integer order integration is replaced with an integration of fractional order 1.25 and by Ladaci and Charef (2006) where the good performance in noise rejection is emphasized.

In the related literature, the absence of methods designed and implemented via fractional differintegration in robust and nonlinear control are visible. The purpose of this paper is to fill this gap to the extent that covers (1) better robustness and noise rejection capabilities than those utilizing traditional integer order operators, (2) non drifting parametric evolution when the essential factor driving the adaptation scheme is noise and (3) better tracking capability and better system response. The three features mentioned above constitute the major results and contributions of the paper.

This paper is organized as follows: In the next section, we give the Riemann-Liouville definitions of fractional operators used throughout the paper, then we introduce the dynamical description of the 2 DOF direct drive robot. In the fourth section, the sliding mode control via fractional order adaptation is given. Simulation results and the concluding remarks constitute the last part of the paper.

2 Fractional order derivative and integral

Given $0 < \beta < 1$, Riemann-Liouville definition of the β-th order fractional derivative operator ${}_0\mathbf{D}_t^{\beta}$ is given by

$$f^{(\beta)}(t) = {}_0\mathbf{D}_t^{\beta} f(t)$$

$$= \frac{1}{\Gamma(1-\beta)} \frac{\mathrm{d}}{\mathrm{d}t} \int_0^t (t-\xi)^{-\beta} f(\xi) d\xi \tag{1}$$

where $\Gamma(\cdot)$ is the Gamma function[1] generalizing the factorial for noninteger arguments. According to this definition, the derivative of a time function $f(t) = t^{\alpha}$ with $\alpha > -1, t \geq 0$ is evaluated as

$${}_0\mathbf{D}_t^{\beta} t^{\alpha} = \frac{\Gamma(\alpha+1)}{\Gamma(\alpha+1-\beta)} t^{\alpha-\beta} \tag{2}$$

Likewise, Riemann-Liouville definition of the β-th order fractional integration operator ${}_0\mathbf{I}_t^{\beta}$ is given by

$${}_0\mathbf{I}_t^{\beta} f(t) = \frac{1}{\Gamma(\beta)} \int_0^t (t-\xi)^{\beta-1} f(\xi) d\xi \tag{3}$$

The material presented in the sequel is based on the above definitions of fractional differentiation and integration as well as the following integration rules describing the integral of a derivative $f^{(\beta)}$ and a constant c.

$${}_0\mathbf{I}_t^{\beta} f^{(\beta)} = f(t) - f^{(\beta-1)}(0) \frac{t^{\beta-1}}{\Gamma(\beta)} \tag{4}$$

$${}_0\mathbf{I}_t^{\beta} c = c \frac{t^{\beta}}{\Gamma(1+\beta)} \tag{5}$$

This paper adopts Riemann-Liouville definition of the fractional differintegration operators among the four fundamental definitions by Riemann-Liouville, Caputo, Grünwald-Letnikov and Kolwankar and Gangal, (See Das 2008). The rationale behind our choice is to approach the stability issues from a more comprehensible viewpoint. In the next section, the control problem is described with a thorough description of the robot dynamics.

3 Plant dynamics and the control problem

In this paper, we consider the following system to visualize the contributions of this paper. The motivation for choosing this system is the nonlinear and coupled nature of differential equations describing the behavior. Furthermore, the adverse effects of noise, large initial conditions and varying payload conditions make the control problem a challenge for conventional approaches.

The dynamics of the robot is given by

$$M(\theta)\ddot{\theta} + V(\theta, \dot{\theta}) = \tau - \eta \tag{6}$$

where $\theta = (\theta_1 \ \theta_2)^{\mathrm{T}}$ vector of angular positions in radians and $\dot{\theta} = (\dot{\theta}_1 \ \dot{\theta}_2)^{\mathrm{T}}$ is the vector of angular velocities in rad/sec. In (6), $\tau = (\tau_1 \ \tau_2)^{\mathrm{T}}$ is the vector of control inputs (torques)

[1] The Gamma function is defined as $\Gamma(\beta) = \int_0^{\infty} e^{-t} t^{\beta-1} \mathrm{d}t$

and $\eta = (\eta_1 \ \ \eta_2)^{\mathrm{T}}$ is the vector of friction forces. The terms in (6) are given below:

$$M(\theta) = \begin{pmatrix} p_1 + 2p_3\cos(\theta_2) & p_2 + p_3\cos(\theta_2) \\ p_2 + p_3\cos(\theta_2) & p_2 \end{pmatrix} \tag{7}$$

$$V(\theta, \dot{\theta}) = \begin{pmatrix} -\dot{\theta}_2(2\dot{\theta}_1 + \dot{\theta}_2)p_3\sin\theta_2 \\ \dot{\theta}_1^2 p_3 \sin\theta_2 \end{pmatrix} \tag{8}$$

where $p_1 = 3.31655 + 0.18648M_p$, $p_2 = 0.1168 + 0.0576M_p$ and $p_3 = 0.16295 + 0.08616M_p$. Here, M_p denotes the payload mass. The details of the plant model can be found in Direct Drive Manipulator R&D Package (1992) and Efe and Kaynak (2000). The constraints regarding the plant dynamics are $|\tau_1| \leq 245$N, $|\tau_2| \leq 39.2$N, and the friction terms are $\eta_1 = 4.9\mathrm{sgn}(\dot{\theta}_1)$ and $\eta_2 = 1.67\mathrm{sgn}(\dot{\theta}_2)$.

The control problem is to force the system states to a predefined and differentiable trajectories within the workspace of the robot. More explicitly $e_1 = \theta_1 - r_1$, $e_2 = \theta_2 - r_2$ and the (integer order) time derivatives of these error terms are desired to converge the origin of the phase space.

4 Sliding mode control through a fractional order adaptation scheme

Theorem 4.1 *Let r_1 and r_2 be continuous and differentiable reference trajectories. Let the switching function for each link be defined by*

$$s_{p,i} = \dot{e}_i + \lambda_i e_i, \quad i = 1, 2, \quad \lambda_i > 0 \tag{9}$$

Let

$$\tau_i = \phi_i^{\mathrm{T}} u_i, \quad i = 1, 2 \tag{10}$$

be the ADALINE based controller of the i-th link with $\phi_i = (\phi_{i,1} \ \ \phi_{i,2} \ \ \phi_{i,3})^{\mathrm{T}}$ being the adjustable parameter set for the i-th controller and $u_i = (e_i \ \ \dot{e}_i \ \ 1)^{\mathrm{T}}$ being the vector signal exciting the i-th controller. Define $\tau_{d,i}$ as a control signal resulting in the desired system response at i-th link, and $\forall i \in \{1, 2\}$ let

$$\left| \sum_{k=1}^{\infty} \frac{\Gamma(1+\beta)}{\Gamma(1+k)\Gamma(1-k+\beta)} (\phi_i^{(\beta-k)})^{\mathrm{T}} u_i^{(k)} \right| \leq \mathcal{B}_{1,i} \tag{11}$$

$$|\tau_{d,i}^{(\beta)}| \leq \mathcal{B}_{2,i}, \quad \forall i \in \{1, 2\} \tag{12}$$

$$|\tau_i| \leq \mathcal{B}_{3,i}, \quad \forall i \in \{1, 2\} \tag{13}$$

The adaptation law

$$\phi_i^{(\beta)} = -\frac{u_i}{u_i^{\mathrm{T}} u_i} \mathcal{K}_i \mathrm{sgn}(\sigma_i) \tag{14}$$

with $\sigma_i := \tau_i - \tau_{d,i}$ drives the parameters of the i-th controller to values such that the plant under control enters the sliding mode, hitting in finite time satisfying

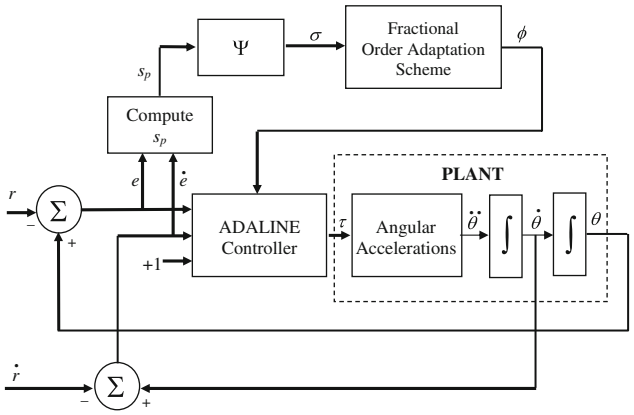

Fig. 1 Block diagram of the control system

$$\frac{\mathcal{K}_i - \mathcal{B}_{1,i}}{\Gamma(1+\beta)} t_{h,i}^{\beta} \leq \frac{|\sigma_i^{(\beta-1)}(0)| + |\tau_{d,i}^{(\beta-1)}(0)|}{\Gamma(\beta)} t_{h,i}^{\beta-1} + |\tau_{d,i}(t_{h,i})| \tag{15}$$

is observed if $\mathcal{K}_i > \mathcal{B}_{1,i} + \mathcal{B}_{2,i}$ is satisfied.

Proof Consider the block diagram of the control system depicted in Fig. 1, where the plant states are assumed to be observable.

The original tuning approach is proposed in Sira-Ramirez and Colina-Morles (1995), where the desired output is available. The integer order version of the problem addressed here is studied in Efe (2002), where the crux of the approach is to extract a quantified error on the applied control signal utilizing the available measurements. In Efe (2002) and Topalov et al. (2007), the map $\Psi(\cdot)$ is a monotonically increasing function of its argument and a common choice for it is a unit function, i.e. $\sigma_i = s_{p,i}$. The practical interpretation of this choice is the adoption of the distance from the switching line as a measure to penalize the control action. That is to say, set of all control signals driving the error vector toward the sliding hypersurface is denoted by $\tau_{d,i}$ and the error on the control signal[2] described by $\tau_i - \tau_{d,i}$ is a monotonically increasing function along the $s_{p,i}$ axis. Such a selection with a tuning mechanism minimizing the value of $s_{p,i}$ naturally forces the emergence of the sliding mode in the conventional sense. □

Define $\Upsilon_i := \sum_{k=1}^{\infty} \frac{\Gamma(1+\beta)}{\Gamma(1+k)\Gamma(1-k+\beta)} (\phi_i^{(\beta-k)})^{\mathrm{T}} u_i^{(k)}$ and check whether the quantity $\sigma_i^{(\beta)} \sigma_i$ for every i is negative

[2] Error on the control signal is naturally not a computable quantity, however, such a quantity can be extracted based on the behavioral properties as considered here and in Efe (2002) and Topalov et al. (2007)

or not. With these expressions, we have

$$\begin{aligned}\sigma_i^{(\beta)}\sigma_i &= \left(\tau_i^{(\beta)} - \tau_{d,i}^{(\beta)}\right)\sigma_i \\ &= \left((\phi_i^{(\beta)})^{\mathrm{T}}u_i\right)\sigma_i + \left(\Upsilon_i - \tau_{d,i}^{(\beta)}\right)\sigma_i \\ &= -\mathcal{K}_i\,\mathrm{sgn}(\sigma_i)\sigma_i + \left(\Upsilon_i - \tau_{d,i}^{(\beta)}\right)\sigma_i \\ &\leq -\mathcal{K}_i|\sigma_i| + |\Upsilon_i|\,|\sigma_i| + |\tau_{d,i}^{(\beta)}|\,|\sigma_i| \\ &\leq (-\mathcal{K}_i + \mathcal{B}_{1,i} + \mathcal{B}_{2,i})|\sigma_i| \\ &\leq 0 \quad \text{Since } \mathcal{K}_i > \mathcal{B}_{1,i} + \mathcal{B}_{2,i} \end{aligned} \tag{16}$$

This proves that the trajectories in the phase space are attracted by the subspace described by $\sigma_i = 0$. Due to the definition in (1), claiming $\sigma_i^{(\beta)}\sigma_i < 0$ for stability is equivalent to the following

$$\sigma_i^{(\beta)}(t)\sigma_i(t) = \frac{\sigma_i(t)}{\Gamma(1-\beta)}\frac{\mathrm{d}}{\mathrm{d}t}\int_0^t \frac{\sigma_i(\xi)}{(t-\xi)^{\beta}}\,\mathrm{d}\xi \tag{17}$$

Obtaining $\sigma_i^{(\beta)}(t)\sigma_i(t) < 0$ can arise in the following cases. In the first case, $\sigma_i(t) > 0$ and the integral $\int_0^t \frac{\sigma_i(\xi)}{(t-\xi)^{\beta}}\,\mathrm{d}\xi$ is monotonically decreasing. In the second case $\sigma_i(t) < 0$ and the integral $\int_0^t \frac{\sigma_i(\xi)}{(t-\xi)^{\beta}}\,\mathrm{d}\xi$ is monotonically increasing. In both cases, the signal $|\sigma_i(t)|$ is forced to converge the origin faster than $t^{-\beta}$. A natural consequence of this is to observe a very fast reaching phase as the signal $t^{-\beta}$ is a very steep function around $t \approx 0$.

Now we must prove that first hitting to the switching function occurs in finite time, denoted by $t_{h,i}$. Evaluate $\sigma_i^{(\beta)}$ utilizing (14) as given below.

$$\sigma_i^{(\beta)} = -\mathcal{K}_i\,\mathrm{sgn}(\sigma_i) + \Upsilon_i - \tau_{d,i}^{(\beta)} \tag{18}$$

Applying the fractional integration operator described in (3) with final time $t = t_{h,i}$ to both sides of (18) one gets

$$\begin{aligned}\sigma_i(t_{h,i}) - \sigma_i^{(\beta-1)}(0)\frac{t_{h,i}^{\beta-1}}{\Gamma(\beta)} &= \frac{-\mathcal{K}_i\,\mathrm{sgn}(\sigma_i(0))}{\Gamma(1+\beta)}t_{h,i}^{\beta} \\ &\quad + {}_0\mathbf{I}_{t_{h,i}}^{\beta}\left(\Upsilon_i - \tau_{d,i}^{(\beta)}\right)\end{aligned} \tag{19}$$

Noting that $\sigma_i(t) = 0$ when $t = t_{h,i}$, multiplying both sides of (19) by $\mathrm{sgn}(\sigma_i(0))$, we have

$$\begin{aligned}&-\sigma_i^{(\beta-1)}(0)\mathrm{sgn}(\sigma_i(0))\frac{t_{h,i}^{\beta-1}}{\Gamma(\beta)} \\ &= \frac{-\mathcal{K}_i}{\Gamma(1+\beta)}t_{h,i}^{\beta} + {}_0\mathbf{I}_{t_{h,i}}^{\beta}(\mathrm{sgn}(\sigma_i(0))\Upsilon_i) \\ &\quad - {}_0\mathbf{I}_{t_{h,i}}^{\beta}(\mathrm{sgn}(\sigma_i(0))\tau_{d,i}^{(\beta)})\end{aligned} \tag{20}$$

Due to the definition given in (3), we have

$$\begin{aligned}{}_0\mathbf{I}_{t_{h,i}}^{\beta}(\mathrm{sgn}(\sigma_i(0))\Upsilon_i) &\leq {}_0\mathbf{I}_{t_{h,i}}^{\beta}|\Upsilon_i| \\ &\leq {}_0\mathbf{I}_{t_{h,i}}^{\beta}\mathcal{B}_{1,i} \\ &= \mathcal{B}_{1,i}\frac{t_{h,i}^{\beta}}{\Gamma(1+\beta)}\end{aligned} \tag{21}$$

Similarly

$$\begin{aligned}&{}_0\mathbf{I}_{t_{h,i}}^{\beta}\left(\mathrm{sgn}(\sigma_i(0))\tau_{d,i}^{(\beta)}\right) \\ &= \mathrm{sgn}(\sigma_i(0)){}_0\mathbf{I}_{t_{h,i}}^{\beta}\tau_{d,i}^{(\beta)} \\ &= \mathrm{sgn}(\sigma_i(0))\left(\tau_{d,i}(t_{h,i}) - \tau_{d,i}^{(\beta-1)}(0)\frac{t_{h,i}^{\beta-1}}{\Gamma(\beta)}\right)\end{aligned} \tag{22}$$

Substituting the results in (21) and (22) into (20), we obtain an inequality given as

$$\begin{aligned}&-\sigma_i^{(\beta-1)}(0)\mathrm{sgn}(\sigma_i(0))\frac{t_{h,i}^{\beta-1}}{\Gamma(\beta)} \leq \frac{-\mathcal{K}_i}{\Gamma(1+\beta)}t_{h,i}^{\beta} \\ &+\mathcal{B}_{1,i}\frac{t_{h,i}^{\beta}}{\Gamma(1+\beta)} - \mathrm{sgn}(\sigma_i(0))\tau_{d,i}(t_{h,i}) \\ &+\tau_{d,i}^{(\beta-1)}(0)\mathrm{sgn}(\sigma_i(0))\frac{t_{h,i}^{\beta-1}}{\Gamma(\beta)}\end{aligned} \tag{23}$$

Straightforward manipulations will lead to the inequality in (15). Clearly, the left hand side of the inequality in (15) is a monotonically increasing function of $t_{h,i}$. On the other hand, the right hand side of the inequality is a monotonically decreasing function of $t_{h,i}$. With these facts, the inequality is satisfied on the interval $t_{h,i} \in (0, \alpha]$, where α is the point of intersection of the two expressions lying on the left and right hand sides of (15). According to this discussion, one can see that $t_{h,i} \leq \alpha$ and particularly for $\beta = 0.5$, we have the following value.

$$\alpha = \left(\frac{|\tau_{d,i}(t_{h,i})| + \sqrt{|\tau_{d,i}(t_{h,i})|^2 + 4\frac{\mathcal{K}_i - \mathcal{B}_{1,i}}{\Gamma(1+\beta)}\frac{|\sigma_i^{(\beta-1)}(0)| + |\tau_{d,i}^{(\beta-1)}(0)|}{\Gamma(\beta)}}}{2\frac{\mathcal{K}_i - \mathcal{B}_{1,i}}{\Gamma(1+\beta)}}\right)^2 \tag{24}$$

Now we turn our attention to the assumptions we made in (11) through (13). Obviously, the assumptions are rather demanding and stringent. The control system presented here would be globally stable if these conditions hold true for the entire course of operation, however, imposing such bounds make the presented design valid only within a local region. Practically, the conditions in (11)–(13) emphasize that the desired sliding regime is achievable after a reaching phase. In other words, the achievability of the task with boundedly

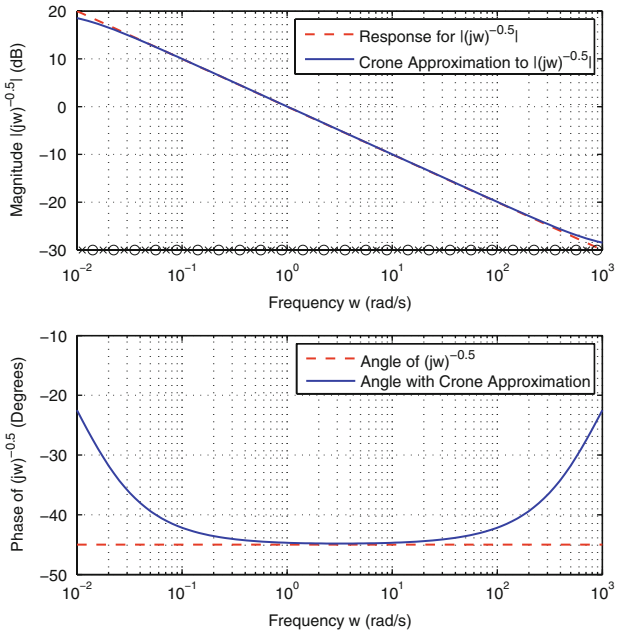

Fig. 2 The approximation to fractional integration operator $s^{-0.5}$ with s being the Laplace variable

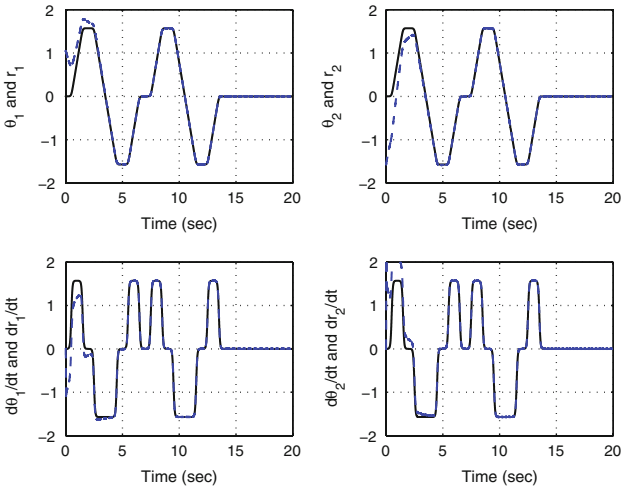

Fig. 3 Reference trajectories and the response of the robot

evolving signals and their derivatives is assumed in (11)–(13).

In the next section, we present an example to demonstrate that the local region mentioned above is indeed large enough to observe highly satisfactory performance. Nevertheless, since one example is not enough to generalize, in the end, we emphasize the usefulness of the presented method.

5 Simulation results

During the simulations, the numerical implementation of the tuning law in (14) is achieved through the use of well known Crone approximation, which prescribes a series of poles and zeros to build a transfer function $k\prod_{i=1}^{N}\frac{1+s/z_i}{1+s/p_i}$ approximating the desired operator spectrally. In Fig. 2, we demonstrate the approximation used in this paper, i.e. for $\beta = 0.5$, we choose $N = 25$ poles and zeros and set the relevant frequency range as $w \in [10^{-2}, 10^{3}]$ rad/s. The gain k is adjusted so that the magnitude is 0 dB at $w = 1$ rad/s, the logarithmic deployment of the poles and zeros are also shown along the horizontal axis of the magnitude plot where the solid line approximates the dashed line having -10 dB/decade slope. Similarly, the phase is around 45 degrees for a significantly wide subband of the chosen frequency region. Expectedly, the quality of the approximation is deteriorated near the limits of the interval, yet one can choose larger frequency bands with higher N at the cost of increasing computational burden.

The presented approach is implemented for the plant introduced in the second section. The system runs for 20 s of time and the reference trajectories shown in Fig. 3 are used. The solid curves represent the reference trajectories while the dashed ones stand for the response of the robot. During the operation, a 5 kgs. of payload is grasped when $t = 2$ s. and released when $t = 5$ s and this is repeated when the robot is motionless at $t = 9$ and 12 s. The manipulator is desired to stay motionless after $t = 15$ s.

It should be noted that the payload scenario is a significant disturbance changing the dynamics of the plant suddenly. Another difficulty is the initial conditions that the ADALINE controllers are supposed to alleviate. Initially, $r_1 = r_2 = 0$, the system is motionless and $\theta_1(0) = \frac{\pi}{3}$ and $\theta_2(0) = -\frac{\pi}{2}$, which are large enough to test the performance of a control scheme. The simulations are carried out with a time step of 2.5 ms and $\mathcal{K}_1 = 1,000$ and $\mathcal{K}_2 = 100$ values are chosen after just a few trials. The sliding lines for both links are set by choosing $\lambda_i = 1$. Beside these, in order to avoid exciting any undesired chattering phenomenon associated tightly with the discontinuous nature of the sign function, we choose $\text{sgn}(\sigma_i) \approx \frac{\sigma_i}{|\sigma_i|+\delta}$ with δ being the parameter determining the slope around the origin. This paper considers $\delta = 0.01$ introducing a very thin boundary layer.

The discrepancies between the reference trajectories and the system response are depicted in Fig. 4, where an exponential convergence is apparent even in the presence of noise corrupting the observed system states and the changes in the system dynamics due to the payload variations.

The behavior in the phase space illustrated in Fig. 5 is another evidence of robustness of the control system and insensitivity to variations in the plant dynamics.

In Fig. 6, the applied control signals are given with the window graphs for better visualizing the initial transient. As expected, the control efforts during the first ten milliseconds have higher magnitudes than what comes later. The effect of

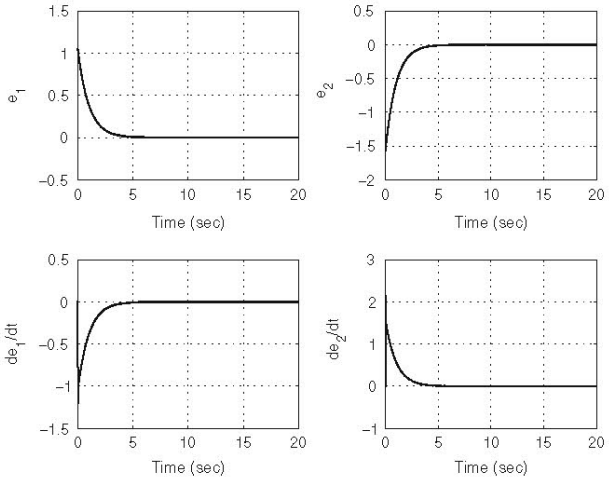

Fig. 4 State tracking errors

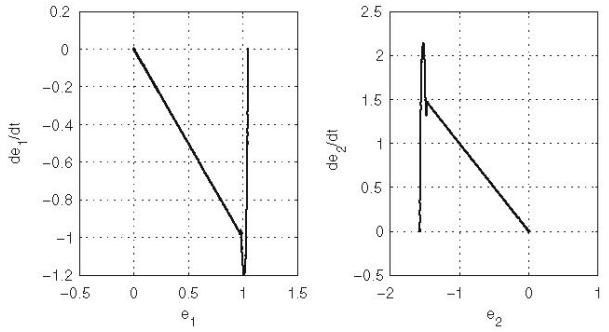

Fig. 5 Behavior in the phase space

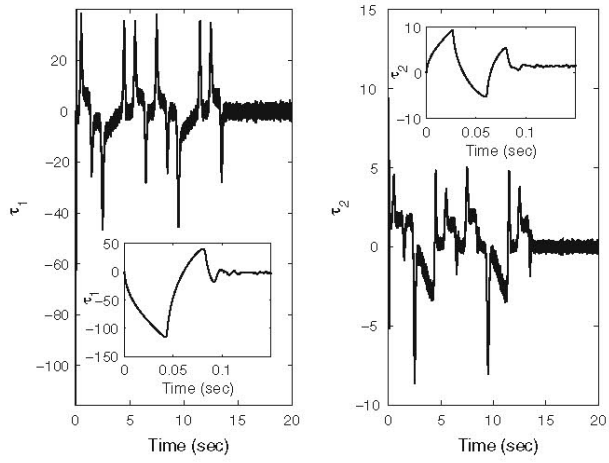

Fig. 6 Applied control signals and their initial transients

noisy observations on the control signal is another conclusion that worths mentioning.

The time evolution of the controller parameters, which are all started from zero, are shown in Fig. 7, where it is clearly visible that after a fast transient, shown also in Fig. 8, the

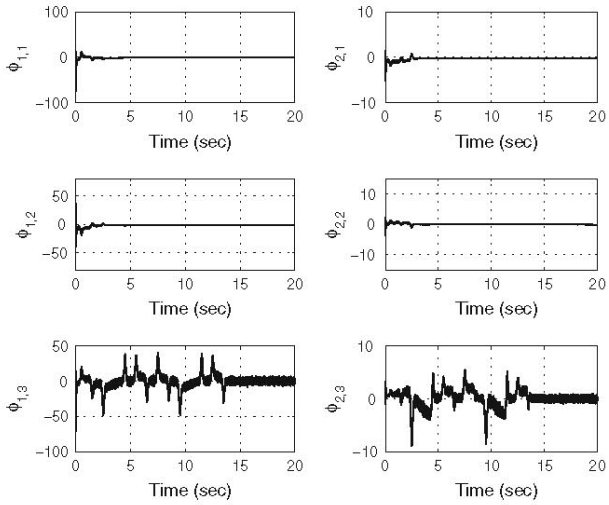

Fig. 7 Time evolution of the controller parameters for base link (*left column*) and elbow link (*right column*)

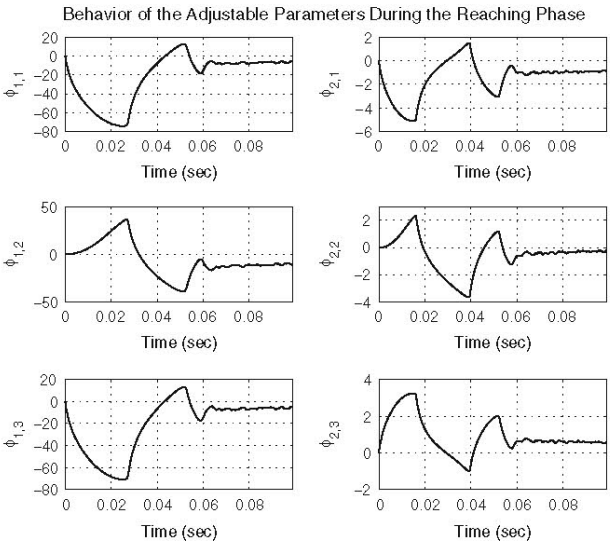

Fig. 8 Time evolution of the controller parameters during the reaching phase. Base link (*left column*) and elbow link (*right column*)

parameters settle down to constant values. If we remember the reference profiles, the system is desired to be motionless after $t > 15$ s, this means that the tuning activity during this time is subject to the effects of noise. That is to say, the system is at a desired state but we would like to figure out how the parameter tuning mechanism functions during this period. The answer to this question is on the Fig. 7, where any possible undesired drift in the controller parameters are suppressed appropriately.

In Figs. 9 and 10, we demonstrate the results obtained with integer order version of the same tuning scheme. In this case, several hits occur before the error vector gets trapped to the sliding manifold. Clearly, a comparison of the cases

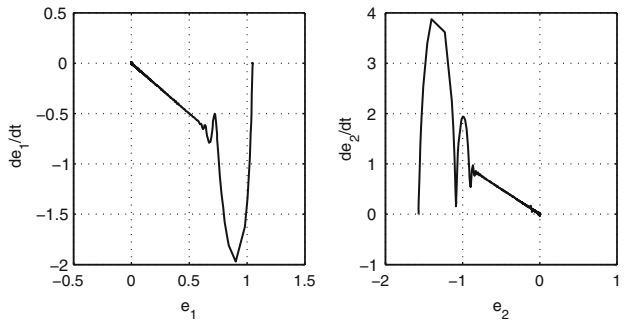

Fig. 9 The behavior in the phase space for the integer case, i.e. $\beta = 1$

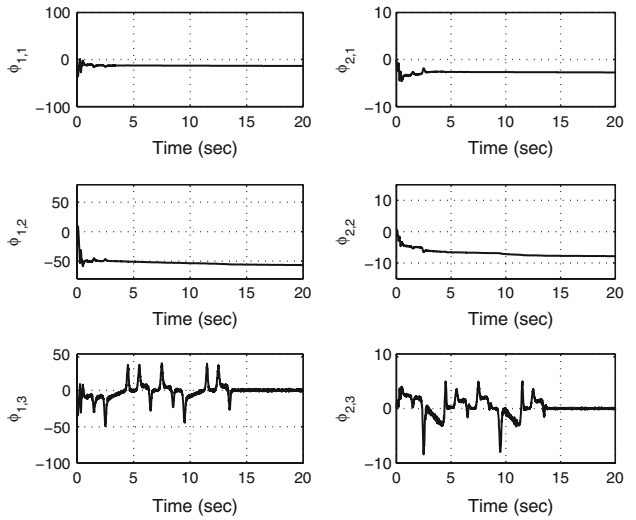

Fig. 10 For the integer case, i.e. $\beta = 1$, the time evolution of the controller parameters for base link (*left column*) and elbow link (*right column*)

$\beta = 1$ and $\beta = 0.5$ suggests using fractional order form of the tuning law as it creates a more attractive sliding manifold than the integer order one. Looking at the parameter evolutions, although in the fractional order case the parameters excited by +1 (bottom subplots in Figs. 7, 10) are seem to be influenced slightly, the drift in the middle subplots of Fig. 10 indicate that the overall system will experience instability in the long run. Since the sampling period is long enough, the variations in the fractional order case are within the tolerable range.

6 Conclusions

In this paper, we propose a fractional order parameter tuning scheme, which was utilized with integer order operators in the past literature. A two degrees of freedom (DOF) planar robot is utilized to justify the claims and a comparison with the integer order version is presented. The presented form of the adaptation law provides

- Better parametric evolution that displays no drifts
- Better tracking capabilities
- Better robustness and disturbance rejection capabilities

than its integer order counterpart, which is only computationally simple.

Briefly, according to the considered application, the fractional order tuning law outperforms the tuning mechanisms exploiting integer order operators.

Future work of the author is to provide a rigorous proof for bounded evolution of the adjustable parameters.

Acknowledgments The Matlab toolbox *Ninteger v.2.3*[3] is used and the efforts of its developer, Dr. Duarte Valério, are gratefully acknowledged. This work is supported by Turkish Scientific Council (TÜBİTAK) Contract 107E137.

References

Åström KJ, Wittenmark B (1995) Adaptive control, 2nd edn. Addison-Wesley, Reading

Das S (2008) Functional fractional calculus for system identification and controls. Springer, Berlin

Direct Drive Manipulator R&D Package, User Guide (1992) Integrated Motions Incorporated, 704 Gillman Street, Berkeley, California 94710, USA

Efe MÖ (2002) A novel error critic for variable structure control with an ADALINE. Trans Inst Meas Cont 24:403–415

Efe MÖ, Kaynak O (2000) A comparative study of soft computing methodologies in identification of robotic manipulators. Rob Auton Syst 30:221–230

Haykin S (1994) Neural networks. Macmillan College Printing Company, New Jersey

Jang J-SR, Sun C-T, Mizutani E (1997) Neuro-fuzzy and soft computing. PTR Prentice-Hall, Englewood Cliffs

Ladaci S, Charef A (2006) On fractional adaptive control. Nonlinear Dyn 43:365–378

Matignon D (1998) Stability properties for generalized fractional differential systems. ESAIM Proc 5:145–158

Momani S, Hadid S (2004) Lyapunov stability solutions of fractional integrodiffrential equations. Int J Math and Math Sci 47:2503–2507

Oldham KB, Spanier J (1974) The fractional calculus. Academic Press, New York

Podlubny I (1998) Fractional differential equations, 1st edn. Elsevier, Amsterdam

Sira-Ramirez H, Colina-Morles E (1995) A sliding mode strategy for adaptive learning in Adalines. IEEE Trans Circuits Syst - I: Fundam Theory Appl 42(12):1001–1012

Topalov AV, Cascella GL, Giordano V, Kaynak O (2007) Sliding mode neuro-adaptive control of electric drives. IEEE Trans Induc Electron 54:671–679

Vinagre BM, Petraš I, Podlubny I, Chen YQ (2002) Using fractional order adjustment rules and fractional order reference models in model-reference adaptive control. Nonlinear Dyn 29:269–279

[3] http://mega.ist.utl.pt/~dmov/ninteger/ninteger.htm