Biomedical Signal Processing and Control 80 (2023) 104410

Contents lists available at ScienceDirect

Biomedical Signal Processing and Control

journal homepage: www.elsevier.com/locate/bspc

Adaptive multi symptoms control of Parkinson's disease by deep reinforcement learning

Behnam Faraji [a,*], Korosh Rouhollahi [b], Saeed Mollahoseini Paghaleh [c], Meysam Gheisarnejad [d], Mohammad-Hassan Khooban [d]

[a] Department of Computer and Electrical Engineering, University College of Rouzbahan, Sari, Iran
[b] Department of Applied Mathematics, Yazd University, Yazd, Iran
[c] Department of Sport Science, Mazandaran University, Babolsar, Iran
[d] Department of Electrical and Computer Engineering, Aarhus University, 8200 Aarhus, Denmark

ARTICLE INFO

Keywords:
Parkinson's Disease (PD)
Deep Brain Stimulation (DBS)
Rigidity
Hand tremor
Deep Deterministic Policy Gradient (DDPG)
Sliding Mode (SM)

ABSTRACT

Parkinson's disease (PD) is one of the really frequent disorders, with hand and head tremors and rigidity being the most common sequelae. Deep brain stimulation (DBS) is a common treatment used to alleviate the symptoms of this disease. This work investigates an ultra-local model (ULM) based on a sliding mode observer (SMO) to simultaneously reduce hand tremor and rigidity. specifically, a deep deterministic policy gradient (DDPG) controller is adaptively designed in the current study to reduce observer estimation error and improve the nonlinear dynamic features of a central neural network (CNN). The DDPG is designed with an actor that produces policy demands and a critic that measures the effectiveness of the actor's policy orders. The offered methodology employs a DDPG-based mechanism to compensate for the shortcomings of the ULM-based SMO. In the present mechanism, training of the weight values of both networks (actor and critic) is by the gradient descent way that relies on the tremor fault's reward return. Finally, the following methodology is analyses by computer simulation in a variety of contexts (robustness and controller performance) and compared to current practices to prove the benefits and adaptability of the procedure with varied models and patients. Additionally, the controllers are implemented in the hardware-in-the-loop (HiL) simulations testbed to validate the performance of the developed scheme's profitability from a realistic perspective.

1. Introduction

Parkinson's disease (PD) is known as a neurological disorder in which the substantia nigra (SN), a component of the basal ganglia (BG), is severely damaged and dopamine (DA) levels have dropped [1]. Whereas DA acts as a neurotransmitter to coordinate the basal ganglia's operation, its destruction or disappearance induces rigidity, shaking hands, bradykinesia, and balance problems [2]. Although tremor has known as the main sign of PD and is easily quantifiable, the basis of tremor is still being debated [3]. There are two hypotheses about what causes tremor: ***i)*** Central mechanisms are involved. This theory implies a central origin for the tremor, such as thalamic cells, BG, or cerebellum. ***ii)*** peripheral feedback pathways. According to this theory, PD tremor is caused by an unstable hyper-excitable long-loop reflex arc [3,4]. In past years, there has been presented a huge spectrum of models by researchers such as: representing the central and peripheral loops by the neural network model [5], The external region of the globus pallidus (GP), and the subthalamic nucleus (STN), as well as its connections, is modeled at the cell level [6], the mathematical model of the physiological and pathological functioning of the BG [7].

The rigidity mechanism, unlike the tremor, is recognized to be attributable to the activation of a long-loop reflex [3]. According to the tremor peripheral concept, someone believes both tremor and stiffness have the same processes [8]; Nonetheless, a few experimental studies have examined the relationship between stiffness and tremor such as employing the F-wave period to increase the reactivity of spinal motor neurons in inflexible Parkinson patients [9], demonstrating the longer F-wave duration in Parkinson's tremor and Parkinson's rigidity [10], modeling the relevance of spinal cord in the development of Parkinson's stiffness and associated motor dysfunctions [3,11], illustrating the

* Corresponding author.
E-mail addresses: behnamfaraji22@gmail.com (B. Faraji), k_rouhollahi@stu.yazd.ac.ir (K. Rouhollahi), s.mollahoseini02@umail.umz.ac.ir (S. Mollahoseini Paghaleh), khooban@ieee.org (M.-H. Khooban).

https://doi.org/10.1016/j.bspc.2022.104410
Received 31 May 2022; Received in revised form 4 October 2022; Accepted 7 November 2022
Available online 20 November 2022

relevance of the tremor and rigidity by mathematical model [12].

Over the last two decades, DBS technology, hardware, and clinical outcomes have all improved dramatically, and DBS is now universally recognized as a scientific proof, therapeutic strategy for patients with a variety of neural circuitry dysfunctions [13]. Because of its lower side effect profile, DBS has gradually gained popularity over ablative therapy for most brain neuromodulation applications. Medication-refractory tremor is one of the most common indications for DBS therapy, with what appears to be the main being Essential Tremor (ET) and PD [14]. The DBS technique involves implanting bilateral electrodes in the BG (subthalamic nucleus, globus pallidus, and ventralis intermedius) [15]. The stimulation waveform used determines the efficacy of the therapy. The amplitude (voltage or current), frequency, and pulse width are the three elements that define the waveform [16].

Up till now, a variety of approaches have evolved in the closed-loop and open-loop manners to address the hand tremor and rigidity separately. For example, adaptive active disturbance rejection control (ADRC) [15], sliding mode control (SMC) [17], single-input fuzzy logic controller (FLC) [1,18], backstepping controller [2], proportional integrative derivative (PID) controller [19], closed-loop controller [20], semi-automated approaches [21]. However, these model-based methodologies have been developed to reduce hand tremor and rigidity individually.

The robust model-based strategies fail to correct for the CNN's nonlinearity due to the inevitable uncertainties in the CNN model, such as dynamic nonlinearity, an absence of modeling, and un-modeled dynamics [22,23]. In the presence of the aforementioned unforeseen occurrences, data-driven approaches may be preferable because they are developed utilizing system input–output (I/O) data and do not require the development of a system model. The most significant data-driven techniques for a model-free recurrent update of control variables are incremental Learning Control [23,24], and Deep Reinforcement Learning (DRL) [25], whereas Virtual Reference Feedback Tuning (VRFT) [26] is a non-iterative data-driven scheme. The data-driven intelligent Proportional Integral Derivative (iPID) [27] has been shown to manage diverse nonlinear systems.

Using the intelligent PI (iPI), it is challenging to achieve the perfect solution and desired efficiency while setting nonlinear feedback parameters [28,29]. To address the challenge, several approaches, including heuristic techniques and fuzzy logic, were used to create the coefficients contained in the nonlinear feedback rule [22,23,27]. Deep deterministic policy gradient (DDPG) has recently risen to prominence in the fields of artificial intelligence (AI) and machine learning (ML) as a strong way of solving high-dimensional issues [30]. The fundamental concept behind DDPG is to provide a performance index for learning a set of agents from a significant quantity of plant data created throughout the decision-making process in order to evaluate a decision-making specification in a specific type of program. The input signal of the system for the advanced closed-loop DBS strategy is the amplitude of the hand tremor. The DBS parameters are then automatically updated as a result of the DDPG algorithm's attempt to estimate the planned controller coefficients online in relation to the tremor rate. Up till the Parkinsonian tremor is reduced, this cycle continuous. Outline the recommended strategy is illustrated in Fig. 1. Generally, the current study makes the following achievements:

1) In response to the aforementioned issues, an intelligent control strategy has been established for reducing hand tremors and rigidity in Parkinson's patients, respectively.

2) An ULM controller was developed in a model-free framework to mitigate simultaneously tremor and rigidity by stimulating the basal ganglia system in the current study. In the ULM controller, an SMO is used to estimate the ULM's poorly understood dynamics. By adopting the input and output of the BG system, the ULM controller was designed without the need for the model identification of the basal ganglia system.

3) The learning capability of DDPG is used to reduce tremor and rigidity by interacting the agent with the BG model. The Actor and Critic of the learning control are trained in a model-free way by introducing a reward function as the optimizing goal.

4) The hardware-in-the-loop (HiL) simulations under various scenarios of the dynamic system were accomplished for real-time analysis of the designed controllers.

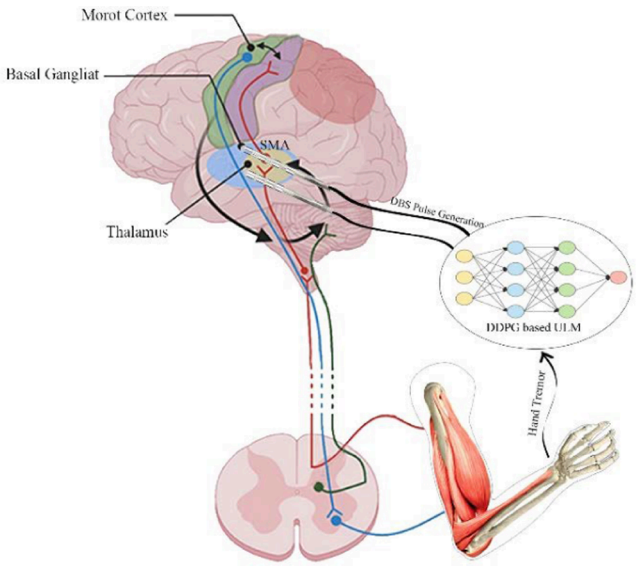


Fig. 1. Outline the recommended strategy.

2. Modelling of hand tremor and rigidity

Based on [3] the dynamic model of hand tremor and rigidity consists of two sections: ***i)*** the agonist and antagonist muscles, as well as the feedback they provide to the spinal cord and ***ii)*** the peripheral and central nervous systems. The central neural part consists of the cortex and BG, which send and receive signals to and from muscles, while the peripheral part consists of the long-loop reflex. In the following, the overall sections and subsections are explained.

2.1. Dynamic model of muscle

As presented in [3,31], the muscle dynamic model is achieved according to the Hill mechanical system, including two springs that symbolize the tendon's and muscular connective tissue's passive characteristics, a length-dependent force generator, and a damper that connects the muscle force to the velocity. Muscle arm formulations from Mains and Soechting are used to simulate a muscle model based on the Hill system. The schematic of the implementing muscle model is illustrated in Fig. 2.

As shown in Fig. 2, The model's input is CNS, and its outcome is hand angular movement. The spinal cord, the spinal cord's action on the

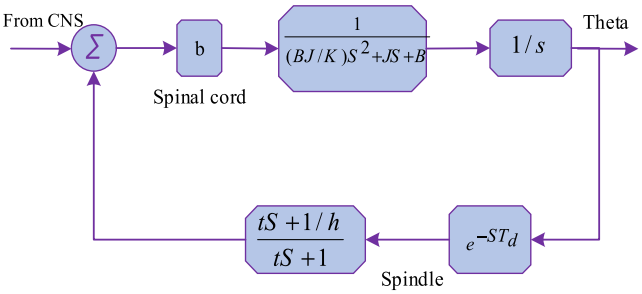


Fig. 2. Illustration of the BG system.

muscle, and delayed feedback from the muscle spindle to the spinal cord are all included in the model. Based on the physiological data, when the CNS activates the agonist's muscle, the antagonist's muscle is at rest, and conversely, due to a feedback mechanism. To implement the mentioned behavior, two gains (g_{af}, g_{anf}) are considered in the spinal cord.

$$\begin{cases} g_{af} = 1 \\ g_{anf} = 0 \end{cases} \Rightarrow (\text{In the presence of agonist's muscle}) \tag{1}$$

$$\begin{cases} g_{af} = 0 \\ g_{anf} = 1 \end{cases} \Rightarrow (\text{In the presence of antagonist's muscle}) \tag{2}$$

2.2. Peripheral and CNS

The peripheral neural element of the model was suggested to be represented by a long loop that begins at the muscle spindle, goes to the dorsal horn of the spinal cord, into the central section of the model, and then returns to alpha motor neurons and the muscle. The supplementary motor area (SMA), the motor cortex, and the BG make up the central part.

2.2.1. Cortex model

The cortex receives signals from the BG and SMA. The amount of brain command to muscles is represented by two parameters: g_a and g_{an}. While the agonist's muscle is recruited, $g_a = 1$ and otherwise $g_a = 0$. Furthermore, g_{an} is associated with antagonist muscle, with $g_{an} = -1$ and $g_{an} = 0$ describing active and inactive states, respectively. A time delay is utilized in the model to depict the alternating commands of the cortex.

The input of SMA to the cortex inhibits the cortex command on muscles in normal patients, but it generates an increase in the cortex command in Parkinson's patients [32]. This behavior is approximated by multiplying the motor command by the SMA input to the cortex, which is for $normalpatients < 1$ and $PD > 1$, respectively. The motor command is expected to be a constant value that represents the cortex command. A sinusoidal signal is sent by the basal ganglia. Normal and PD people have varying amplitudes and frequencies of this signal.

2.2.2. Basal ganglia model

Physiological and pathological tremors are two types of tremors. Physiological tremor is a low-amplitude oscillation that may be detected using appropriate recording methods in almost all normal persons. This tremor has a frequency of 8–12 Hz which is double that of Parkinson's tremor. Here, the basal ganglia are assumed to be sinusoidal.

$$g = sin\omega t$$

$$\omega = 2\pi f \tag{3}$$

$$G_{(s)} = \frac{\omega}{s^2 + \omega^2} = \frac{2\pi f}{s^2 + (2\pi f)^2} \tag{4}$$

In which f (frequency) is proportional to the quantity of dopamine neurotransmitter. Where $f = 5HZ$ and $f = 10HZ$ describe the PD and normal states, respectively.

$$\begin{cases} f = 5Hz \rightarrow \omega = 10\pi = 31.4 \\ G_{(s)} = \dfrac{31.4}{s^2 + (31.4)^2} = \dfrac{31.4}{s^2 + 986} \Rightarrow (PDstate) \end{cases} \tag{5}$$

$$\begin{cases} f = 10Hz \rightarrow \omega = 20\pi = 62.8 \\ G_{(s)} = \dfrac{62.8}{s^2 + (62.8)^2} = \dfrac{31.4}{s^2 + 3944} \Rightarrow (Normalstate) \end{cases} \tag{6}$$

The cortex provides input to the BG, whereas the BG provides output to the cortex. The source block in the simulation represents the source of tremor in PD, and it is assumed to be a constant value that rises as the disease advances. The value of this source is 0 in a typical person.

2.2.3. Supplementary motor area model

In healthy people, SMA has an inhibiting effect. Nevertheless, in those with Parkinson's disease, this inhibitory action weakens, and therefore the excitability of the long-loop reflex rises. By altering the long-loop reflex excitability, the SMA model can adjust the amount of the motor command. There are three parts to the model: the absolute function, the saturation function, and the gain. By adjusting the gain, the SMA output can be improved. Normal people have a saturation function of 0.5 −1, while the saturation function value for PD's people is between 1 and 2.

2.3. Final model

The overall structure of the simulated system is depicted in Fig. 3. As depicted in Fig. 3, the model is made by connecting Section A and Section B. The system's output is expected to be hand angular displacement. Choosing $g_a = 1$ and $g_{an} = 1$ at the same time results in co-contraction, which is a type of muscle rigidity. The level of rigidity increases as the motor command increases. The motor command is raised as the SMA output is increased. As a result, it can be concluded that the amount of SMA output in the co-contraction state is a reliable indicator of rigidity. The nominal values utilized in the described dynamic model are presented in Table 1 [3].

3. Design of the Model-Free DDPG intelligent controller based on SMO

The concept of model-free controllers is not novel, and it encompasses a wide range of control systems that rely as little as possible on the plant's mathematical models. Fuzzy logic and neural networks are examples of these methodologies used in model-free controllers. Another form of the model-free algorithm relies on the ULM notion, which has recently been established in the relevant literature and has been effectively utilized in numerous applications [30]. It should be noted that model-free does not imply the absence of a dynamic model; instead, due to uncertainties and nonlinearities in dynamics, constructing an accurate model representing system dynamics is a challenging and time-consuming effort.

In this part, we suggest a novel model-free DDPG intelligent PI (DDPGiPI) framework with SMO. Three sub-components make up the recommended structured controller: ***A)*** A model-independent feedback method based on the ULM is used to optimize the existing architecture, provide the requisite required specifications, and minimize the output high order derivative. ***B)*** A SMO is used to detect abnormal behaviors using information from the CNN system's control I/O data. ***C)*** A DDPG sub-controller to eliminate the SM observer's assessment mistake by improving the overall controller's effectivity.

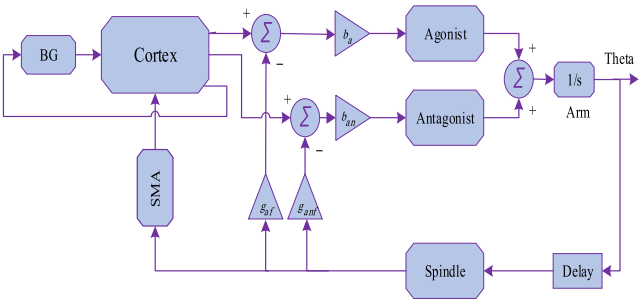


Fig. 3. Dynamic model of tremor and rigidity.

Table 1
Configuration of The Dynamic Model.

Parameters	values	Parameters	values
Muscle stiffness,k	$50Nm$	Delay of spindle,T_d	$0.02s$
Viscous damping,B	$2Nms$	Time constant,τ	$\frac{1}{300}s$
Spinal cord,b	0.01	Constant value,h	5
b_a	0.01	b_{an}	0.01
Forearm inertia about the elbow joint,J			$0.1kgm^2$

3.1. Design of intelligent PI controller

Only the system's dynamic behavior is expected to be accurately simulated within its working range using ordinary differential approaches that may have nonlinear and time-varying properties. The ULM form of a single-input/single-output (SISO) system can be characterized by an unknown ordinary differential, as formulated by the implicit function theorem [33].

$$F\left(t, O, \dot{O}, \cdots, O^{(l)}, I, \dot{I}, \cdots, I^{(k)}\right) = 0 \tag{7}$$

where y (derivative order of the output variable) and u (input variable) illustrate by O and I, and F signifies a sufficiently smooth function of its arguments. Defining that for an integer variable follows: $\sigma, 0 < \sigma < 1, \partial F/\partial O^{\sigma}$. The input/output characteristic can be approximated by a totally unknown or partially known finite-dimensional differential equation according to the implicit function theorem:

$$O^{\sigma} = \Theta\left(t, O, \dot{O}, \cdots, O^{(\sigma-1)}, O^{(\sigma+1)}, \cdots, O^{(l)} I, \dot{I}, \cdots, I^{(k)}\right) \tag{8}$$

It's worth noting that the unknown functional, $\Theta(0)$ in Eq. (9) does not have to be linear or time-invariant. By assigning, $\Theta = \Psi + \Lambda I$, the model-independent scheme's instantaneous formulation can be written described as [34]:

$$O^{\sigma}(t) = \Psi(t) + \Lambda I(t) \tag{9}$$

All model-free phenomena and turbulence like nonlinearity and friction are represented by ψ, and $\Lambda \in \mathbb{R}$ is a parameter of nonphysical design. Whenever $\sigma = 1$ the feedback controller's control law is as follows:

$$I(t) = \frac{1}{\Lambda}\left(- \hat{\psi} + \dot{O}^{*} + k_p e_1 + k_i \int e_1 \right) \tag{10}$$

where kp and ki represent the equivalent usual proportional (P) and integral (I) parameters, O^{*} is the desired reference, e_1 describe the error function, and $\hat{\psi}$ denotes the approximate of $\psi(t)$.

Remark 1. *Eq. (9) is used to create the ULM of the examined model. In these kinds of systems,$\psi(t)$ makes no distinction between nonlinearities and unknown system constituents. Because Eq. (9) is only applicable for a finite time interval, the numeric element $\hat{\psi}(t)$ must be modified through each sampling time. The SM observer is considered to predict the coefficient $\hat{\psi}(t)$ and send feedback to the DDPGiPI platform.*

3.2. Formulating the SM observer

The SMO is commonly used in the literature to approximate a system's unknown dynamics. The popularity of this form of an observer can be attributed to the simplicity, design flexibility, and robust control structure that somewhat compensates for the observer's reliance on the system model. Fig. 4 depicts the construction of the SMO based on the ULM controller. In this application, the iPI is employed as a feedback controller to keep the model result stable. The SM o is used to estimate and return to the controller the ULM's undesired perturbations. The estimated value was derived using the control I/O information, $I(t)$ and $O(t)$, it can be calculated as:

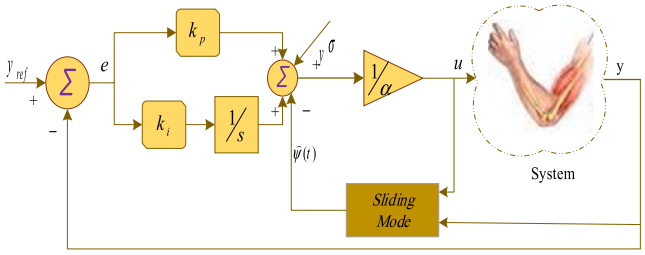


Fig. 4. Structure of the suggested controller.

$$\dot{\hat{O}} = \xi sgn(O - \hat{O}) + \Lambda I \tag{11}$$

The sliding mode parameter has been specified by $\xi sgn(O - \hat{O})$, the estimate of $\hat{O}$ represent by O. The error equation by defining $e_2 = O - \hat{O}$ and deducting Eq. (11) from Eq. (9) can be written as:

$$\dot{e}_2 = \psi - \xi sgn(O - \hat{O}) \tag{12}$$

3.3. DDPG mathematical

Reinforcement learning (RL) is a method of determining the optimum way for an intelligent agent to sense its environment, E, while increasing the agent's reward over time. The interaction process for demonstration and observation can be expressed as a Markov decision process (MDP) under the Markovian characterization of the underpinning environment [35].

i) *State-space*: evaluating the limited set of states in a particular situation., $s \in \mathbb{S}$.

ii) *Action space*: the agent's limited set of operations that will result in a transfer from state s_t at time t to state s_{t+1} at time $t+1$, $(a \in \mathbb{A})$.

iii) *Reward*: At every step in the process, direct feedback from the environment is generated in response to an action performed, $r \in \mathbb{R}$.

iv) *Transition probability distribution*: the possibility that completing action a will cause the state s at time t to shift to the state s' at time $t+1$, $\mathbb{P}(s'|a, s)$.

v) *Policy*: indicates the behavior of the agent that transfers a state s_t to a probability distribution over the control actions a_t, $\pi(a|s)$.

The overall discounted reward per state is specified by $G_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k}$ to formulate the RL equation, where $\gamma = (0, 1]$ specifies the discount factor. The primary goal is to improve the discount value, which follows.:

$$J = \mathbb{E}_{r_i s_i \ E, a_i \ \pi}[G_1] \tag{13}$$

A state value function (V^{π}), according to (14) for a certain policy (where $\pi = a_t$), is a measure of total discounted reward G_t to each $s \in \mathbb{S}$.

$$V^{\pi}(s) = \mathbb{E}_{\pi}[G_t | s_t = s] \tag{14}$$

The Bellman equation can be used to recursively describe Eq (13):

$$V^{\pi}(s) = \mathbb{E}_{\pi}[r_t + \gamma V^{\pi}(s_{t+1}) | s_t = s] \tag{15}$$

Correspondingly, for the state value function that begins at the return, a breakdown of the action-value function is as follows:

$$Q^{\pi}(s, a) = \mathbb{E}_{\pi}[G_t | s_t = s, a_t = a] \tag{16}$$

The obtained function from recursive equation and Bellman equation can be written as:

$$Q^{\pi}(s, a) = \mathbb{E}_{\pi}[r_t | \gamma Q^{\pi}(s_{t+1}, a_{t+1}) | s_t = s, a_t = a] \tag{17}$$

The optimal policy $\pi^{*} = argmax_a Q^{*}(s, a)$ can be obtained from the action-value function, where the Bellman formula exploiting Q can be ideally met.

DDPG is well-known in the field of RL algorithms as one successful implementation of neural networks (NNs) to the RL framework. Because

the above technique's action-state pair is limited to discrete time, two deep neural networks (NNs) are constructed to incorporate the aforementioned strategy into continuous situations. In this technique, two deep NNs namely critic and actor are considered, which both functions $Q(s_t, a_t)$ and $\mu(s_t)$ are directly updated through $Q(s_t, a_t|\theta^Q)$ and $\mu(s_t|\theta^\mu)$. Where θ^Q and θ^μ are symbolized by the coefficients of the critic and actor NNS, respectively. The stochastic gradient descent technique is used to update the critic's coefficients in order to minimize a loss function:

$$\mathscr{L}(\theta^Q) = E_{(a,a)}\left[\left(Q(s_t, a_t|\theta^Q) - y_t\right)^2\right] \tag{18}$$

$$y_t = r_t(s_t, a_t) + \gamma Q\left(s_{t+1}, \mu(s_t|\theta^\mu)\theta^Q\right) \tag{19}$$

The coefficients of the actors θ^μ can be modified based on the following policy gradient:

$$\begin{aligned}\nabla_{\theta^\mu} J^{\theta^\mu} &\approx \mathbb{E}_{s_t\ \rho^\beta}\left[\nabla_{\theta^\mu} Q(s, a|\theta^Q)|_{a=\mu(s|\theta^\mu)} \nabla_{\theta^\mu}\mu(s|\theta^\mu)\right] \\ &= \mathbb{E}_{s_t\ \rho^\beta}\left[\nabla_a Q(s, a|\theta^Q)|_{a=\mu^\theta(s)} \nabla_{\theta^\mu}\mu(s|\theta^\mu)\right]\end{aligned} \tag{20}$$

Where ρ symbolizes the discounted distribution and β represents the specific policy for the current policy.

A reply buffer and the target NNs are also used to further improve the efficiency and resilience throughout training, it represents by the deep Q network (DQN) algorithm. Each time step's experience tuple $e = (s_t, a_t, r_t, s_t+1)$ is kept in an R-sized experience memory $D = \{e_1, e_2, \ldots, e_R\}$ using the reply buffer technique. A mini-batch of tuples is randomly selected from the memory R at each step of the training process. For the actor and critic NNs, two extra NNs $Q'(s, a|\theta^{Q'})$ and $\mu'(s|\theta^{\mu'})$, also known as target networks, are used to avoid the DDPG learning being unstable. Function (21) softly update the weight coefficients of the target networks $\theta^{Q'}$ and $\theta^{\mu'}$ [35]. In addition, for exploration purposes, an exploration noise N based on the Laplace process $N(x|b)\ \frac{1}{2b_t}\exp\left(-\frac{x}{b_t}\right)$ [36] is applied to exiting actor actions $(a_t = \mu(s_t|\theta^\mu) + N)$. The DDPG technique outline is depicted in Fig. 5.

$$\begin{cases}\theta^{Q'} \leftarrow \tau\theta^Q + (1-\tau)\theta^{Q'} \\ \theta^{\mu'} \leftarrow \tau\theta^\mu + (1-\tau)\theta^{\mu'}\end{cases} \quad \tau \in (01] \tag{21}$$

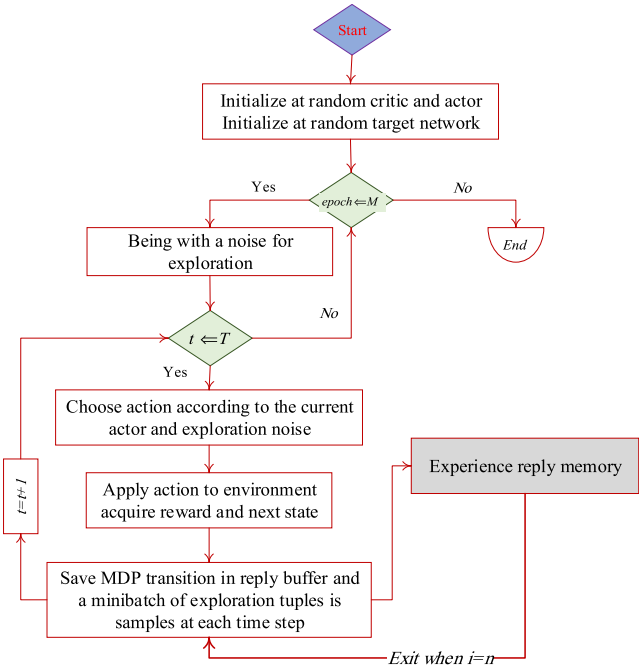


Fig. 5. Diagram of the DDPG algorithm.

3.3.1. Deep Deterministic Policy Gradient Integration in ULM Control. Fig. 6 depicts the updating configuration of ULM parameters using DDPG with the Actor-Critic structure. In Fig. 5 the CNN is the environment in which the DDPG agent interacts with the system and gathers information to discover the best policy. By increasing a reward signal rt, the DDPG action signal, which is created from the Actor NN, is applied to the system to remove the SM observer mistake. As a result, the DDPG-based ULM controller's control law can be stated as follows:

$$I(t) = \frac{1}{\Lambda}\left(-\hat{\psi} + \dot{O}^* + k_p e_1 + k_i \int e_1\right) + I_{DDPG} \tag{22}$$

The purpose of the DDPG is to reduce the error between the reference and actual value by training the Actor weights, θ^μ, and Critic weights, θ^Q, parameters. The Critic analyzes the quality of the Actor policy control for each instant reward, whereas this Actor provides a supplementary control action to compensate for the observer tracking error. The learning technique of the deep NNs for the presented dynamic model in Section II follows:

i) The input signals to the Actor-network are tremor and derivative of tremor, as well as their derivative in time step t, $\left\{Tremor, \left(\frac{d_{tremor}}{dt}\right)\right\}$, and an action $\mu(st|\theta^\mu)$ is produced to eliminate observer tracking.

ii) The Critic-network takes the state vector s_t and the action $\mu(st|\theta^\mu)$ and returns an approximation Q-value $Q(s_t, \mu(st|\theta^\mu)|\theta^Q)$.

The actor NN for the particular dynamic system receives *tremor* and d_{tremor} as the system states (s_t) and then generates two control signals $\{dk_p, dk_i\}$ to modify the ULM gains as $k_p = k_{p0} + dk_p$ and $k_p = k_{i0} + dk_i$. Here, k_{p0} and k_{i0} denote the initial control gains. As the DDPG agent iterates with the dynamic system, the critic NN accepts the system states s_t and the regularity signals $\{dk_p, dk_i\}$, and emits a reward $\{r_t\}$. The weights of the coefficients of the DDPG agent NNs will be organized in such a manner that suppress the tremor value under this set-up of actor and critic networks. The reward function for the DDPG algorithm for the specific application of regulating the output system is formulated as:

$$r_t = \frac{1}{(tremoroutput)^2} \tag{23}$$

The DDPG agent generates control signals according to the received reward signal to suppress the output model and promote system stability. For every update, mini-batch samples of experience tuples are drawn at random from the replay buffer R. The retraining weight variables and biases of Actor and Critic are reset for every mini-batch.

Remark 2. *There is a possibility of divergence while updating the actor and critic networks and exchanging the weight coefficient values for the components at time steps t (current) and t+1 (goal). Small changes in the network weights cause huge oscillations in the coefficients, which is the cause of the*

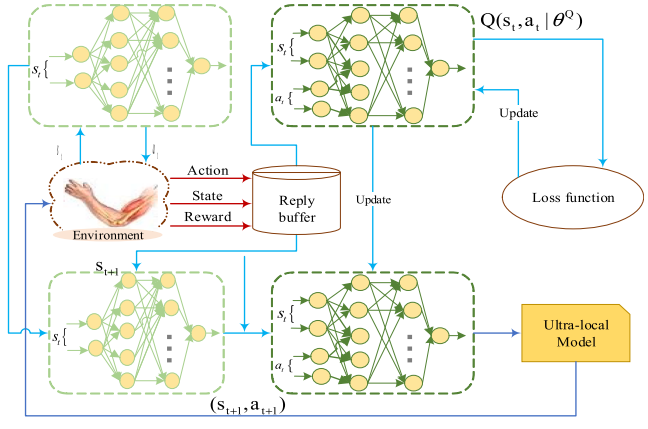


Fig. 6. Indicates the updating configuration of parameters by the DDPG technique.

instabilities. In addition to the core networks, two additional networks consist of the actor target and the critic target are implemented in the DDPG to prevent the learning of the DDPG from becoming unstable. The target networks will adhere to the trained networks if the following soft updates are used.

4. Simulation results

The simulation described in Fig. 2 is integrated with the recommended control method achieved by the structure of Fig. 3 to assess the efficacy of the proposed controller. The recommended closed-loop model-independent controller is carried out in MATLAB/SIMULINK environment to alleviate hand tremor and rigidity in Parkinson's patients at the same time. In the considered scenarios, the offered closed-loop system functionality in decreasing system output is proven, and it compares to other systems such as ULM and PID. The DDPG hyperparameters for the configuration of the suggested strategy are considered as: $\gamma = 0.99$, $\lambda = 0.0001$, and $\tau = 0.01$.

Some testing is carried out by the hardware-in-the-loop (HiL) relying on the OPAL-RT platform for a real-time simulation of the developed strategies. The HiL setup's purpose is to consider the effects of faults and delays on the dynamic system, which are not included in the MATLAB environment's offline simulations. Fig. 7 shows a real-time assessment of the HiL platform for the dynamic model system's designed controllers.

Scenario I: *Controller performance in suppression hand tremor and rigidity based on OPAL-RT platform (real-time examinations):*

In order to assess the proposed controller capability in suppression of hand tremor and rigidity, the closed-loop DBS simulation was performed using the nominal values furnished in Table 1. The output performance of controllers, SMC, ULM, and DDPG-based ULM based on the implementation of the HiL environment for controlling hand tremor and rigidity are presented in the following. Fig. 8 illustrates the tremor output for the various strategies. Additionally, the efficiency of the strategies in suppression rigidity in various states such as co-contraction, in the presence of agonist and antagonist are depicted in Fig. 9 and Fig. 10. From Fig. 8, Fig. 9, and Fig. 10, it can be concluded that the DDPG based ULM structure outperforms the other two controllers in terms of tremor and stiffness mitigation simultaneously.

Scenario II: *Correlation and Robustness evaluation (simulation analysis):*

As previously stated, rigidity and tremor are two sides of a coin. As a result, increasing tremor leads to raising rigidity and vice versa. The outputs of the system have been displayed in Figs. 11 and 12 when SMA increased [16] to show the correlation between rigidity and tremor. It can be seen in Figs. 11 and 12, that increasing SMA causes elevate two

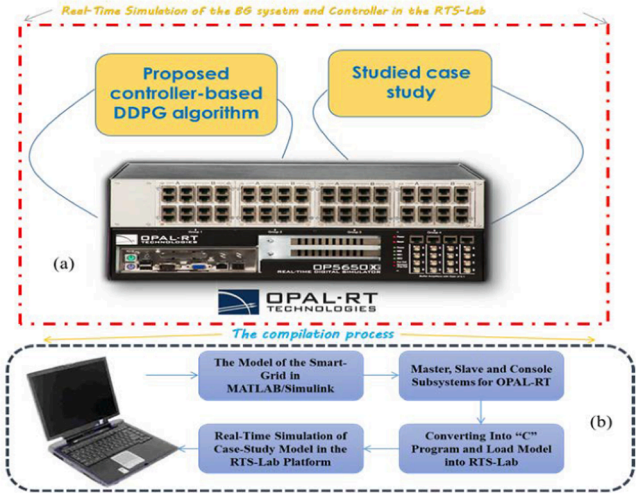


Fig. 7. The real-time experimental setup.

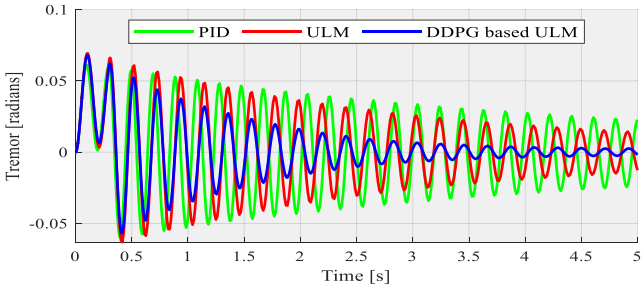


Fig. 8. Tremor output.

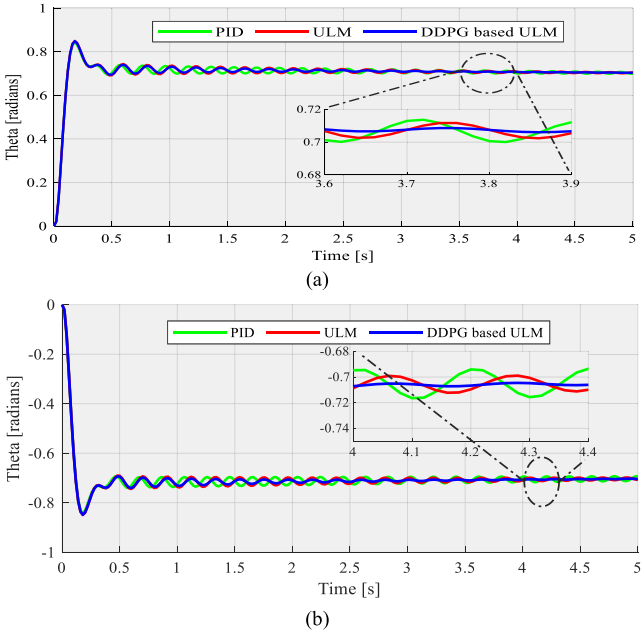


Fig. 9. Rigidity output a) Agonist b) Antagonist.

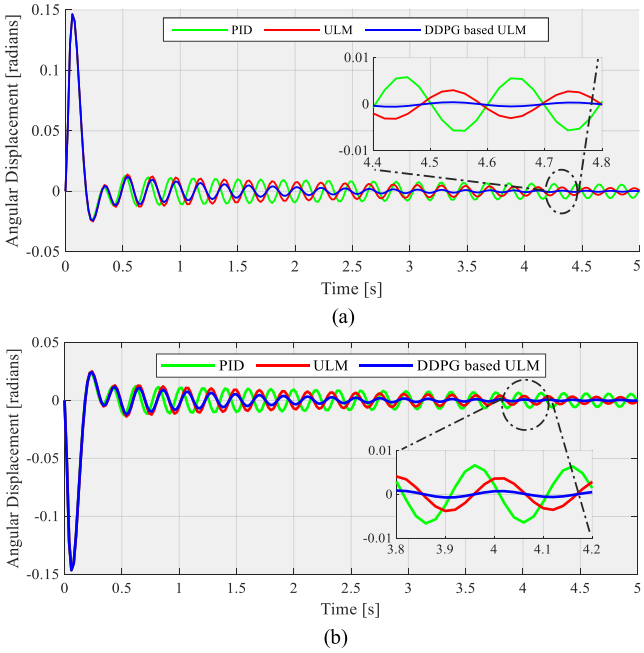


Fig. 10. Co-contraction in the presence a) Agonist b) Antagonist muscles.

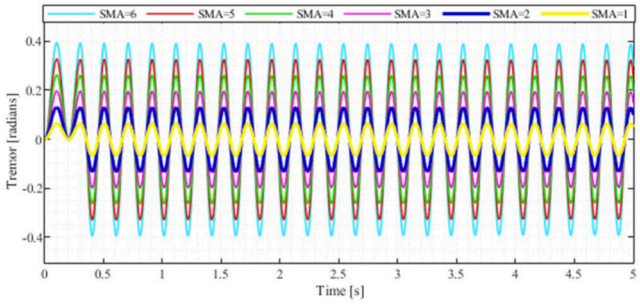
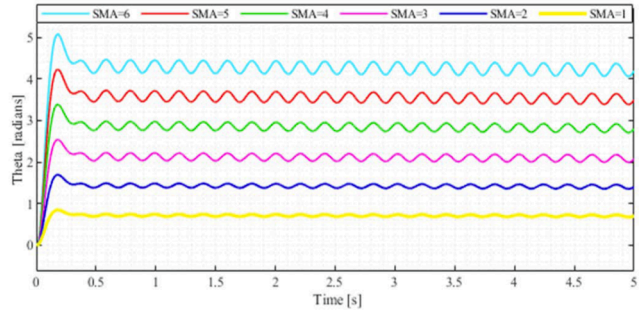
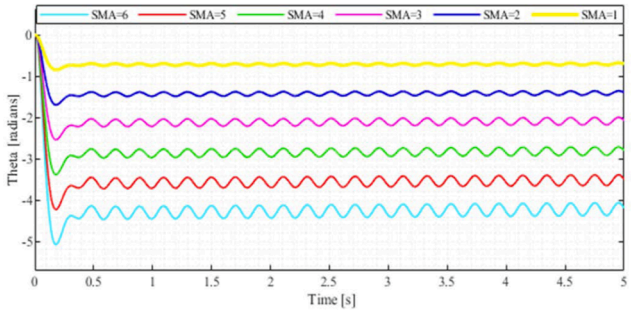


Fig. 11. Tremor output.



(a)



(b)

Fig. 12. Rigidity output a) Agonist b) Antagonist.

signs at the same time, which is equal to severe PD. To demonstrate the robustness ability of the recommended technique in dealing with variation parameters, the gain of SMA oscillated among [16], and the performance index of $J = \int_0^\infty t.tremor^2.dt$ for the tremor, agonist, and antagonist muscles are furnished in Table 2, Table 3, and Table 4, respectively. Furthermore, the outcomes of the approaches under SMA variation are shown by bar charts in Fig. 13, Fig. 14(a), and 14(b). The

Table 2
Performance Index of Controllers Under SMA Variation (Tremor).

SMA	Controllers		
	PID	ULM	DDPG based ULM
1.5	0.0189	0.0098	0.0025
2	0.0336	0.0173	0.0044
2.5	0.0522	0.027	0.0069
3	0.0752	0.0388	0.0104
3.5	0.1022	0.053	0.0138
4	0.1312	0.0687	0.0179
4.5	0.1689	0.0871	0.0218
5	0.2089	0.1078	0.0279
5.6	0.2479	0.1297	0.0346
6	0.2975	0.1546	0.0414

Table 3
Performance Index of Controllers Under SMA Variation (Agonist Muscle).

SMA	Controllers		
	PID	ULM	DDPG based ULM
1.5	14.4400	13.8590	10.8651
2	25.9869	25.0829	19.6750
2.5	40.8794	39.5838	31.1029
3	59.1281	57.4270	45.0458
3.5	80.7317	78.5417	61.6092
4	105.7325	102.9578	80.7848
4.5	134.0676	130.7236	102.4734
5	165.7210	161.6631	126.8455
5.6	200.8080	196.0525	140.8944
6	239.2046	233.6716	167.3781

Table 4
Performance Index of Controllers Under SMA Variation (Antagonist Muscle).

SMA	Controllers		
	PID	ULM	DDPG based ULM
1.5	14.4400	13.8590	10.8650
2	25.9868	25.0827	19.6748
2.5	40.8792	39.5833	31.1025
3	59.1278	57.4262	45.0451
3.5	80.7311	78.5405	61.6082
4	105.7316	102.9561	80.7833
4.5	134.0663	130.7212	102.4713
5	165.7194	161.6601	126.8428
5.6	200.8101	196.0487	140.8944
6	239.2021	233.6669	167.3781

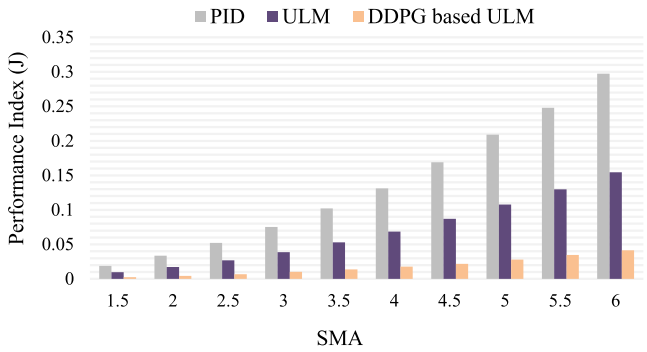


Fig. 13. Performance index *(J)* of tremor according to variation SMA.

recorded data of the Tables and figures verified the superior functional of the DDPG-based ULM in the suppression of hand tremor and rigidity at the same time compared to the other methodologies.

5. Results analysis and justification of Choosing the DDPG-Based ULM technique

5.1. Reasons for implementing a supplemental controller based on DDPG

Several studies in recent literature indicated that model-free controllers, such as ULM, are more practical than traditional counterparts due to their reliance on input/output data and lack of the need for model identification. However, the inability of these controllers to perform online optimization and adaptation precludes them from achieving optimal performance in dealing with noise and other obstacles. Regrettably, tuning the settings of the specified controllers is a challenging operation because the DBS system is frequently employed in sensitive operations and has issues such as instability and nonlinearity.

The motivation for the presented framework is to improve the basic ULM controller by incorporating a DDPG-based supplementary

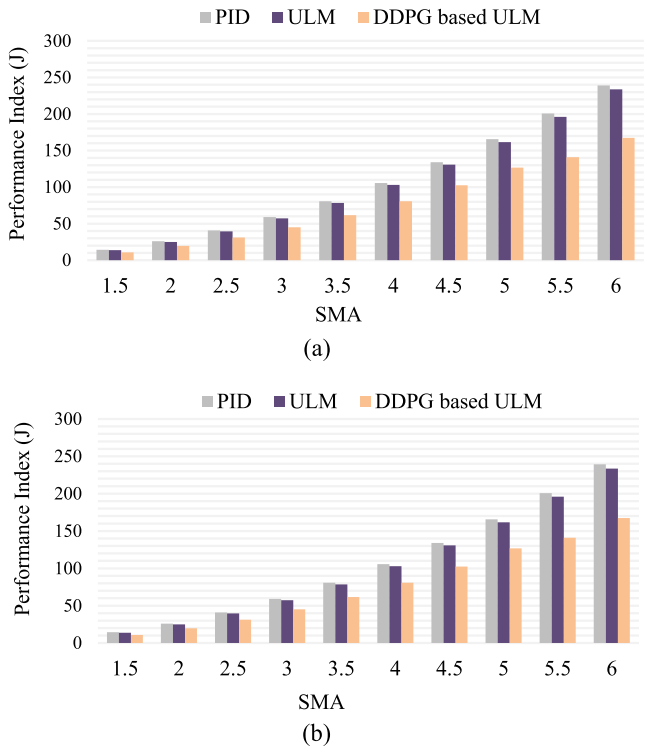


Fig. 14. Performance index *(J)* of rigidity according to variation SMA, a) Agonist, and b) Antagonist.

component with adaptable capabilities. A ULM controller is utilized in particular to ensure fundamental control of the nonlinear dynamic model problem. The DDPG controller was created to work in conjunction with the principal controller in the process of adapting to the handled model's features.

5.2. *The recommended architecture's supremacy*

The following are the characteristics of the proposed adaptive controller over existing controllers for simultaneous regulation of tremor and rigidity:

i) Unlike conventional model-based approaches like linear matrix inequalities (LMI) and model predictive control (MPC) [18, 40], the proposed system is designed in an un-modeled context and will not require modeling recognition.

ii) Optimal tracking using traditional approaches (metaheuristics and neural network algorithms) is typically restricted to particular cycle durations due to optimization and adaptation limits. The controller actions are adjusted at every sampling interval depending on the model's i/o data using the proposed technique to adaptively identify the appropriate control acts for the provided reward.

iii) Rather than using the normal RL technique, Q-learning replaces the initial Q-table in DDPG with a deep neural network. As a result, rather than generating the Q-Table, the deep neural network provides a rapid approximation of the Q-values, resulting in significant processing savings.

iv) The DQL is rendered ineffectual in continuous state operations since it is implemented with only a few feasible acts. The approach can be extended to DDPG on a range of continuous control spaces by using a parametric 'actor'.

6. Conclusion and future works

The outline of the current paper offers an adaption pattern for treating hand tremors and rigidity at the same time via an online update nonlinear strategy (DDPG-based ULM). The DDPG algorithm's learning capability is integrated with the ULM as a base controller and SM observer in the provided structure, which is advantageous to the control of the output symptoms by responding to the instabilities induced by CNN. The controller mechanism is made up according to the muscle model, peripheral and CNS models, also the central part that contains three sub-models: cortex, BG, and SMA. A ULM controller is best constructed in this fashion to produce the principal dynamic control actions for decreasing the output. An SMO is also incorporated to predict unknown dynamics. A unique DDPG with the actor-critic plant is flexibly constructed to supplement the controller with further learning abilities in order to properly handle the system. The training of the actor-critic is via system information from the patient's hand, and they then automatically compensate for uncertainty to achieve a stable outcome. Subsequently, a HIL simulator was used to evaluate the proposed framework's appropriateness in a real-time testbed. The experimental simulation results demonstrated that the suggested model-free controller is superior to the other controllers. The performance index results show that using the suggested model-independent method rather than other methods improves tremor and rigidity, particularly in terms of the various coefficients (sever) and robustness, which is significant from a biomedical engineering point of view. The scope of this study could be broadened in the future to include the use of actual data to investigate the effects of therapies in the context of Parkinson's disease. For design and analytic goals, other PD symptoms, such as head tremor, also can be explored.

CRediT authorship contribution statement

Behnam Faraji: Conceptualization, Writing – original draft. **Korosh Rouhollahi:** Methodology. **Saeed Mollahoseini Paghaleh:** Formal analysis, Validation. **Meysam Gheisarnejad:** Software. **Mohammad-Hassan Khooban:** Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

[1] M. Gheisarnejad, B. Faraji, Z. Esfahani, M.-H. Khooban, A Close loop multi-area brain stimulation control for Parkinson's Patients Rehabilitation, IEEE Sens. J. 20 (4) (2019) 2205–2213.
[2] K. Rouhollahi, M.E. Andani, S.M. Karbassi, I. Izadi, Designing a robust backstepping controller for rehabilitation in Parkinson's disease: a simulation study, IET Syst. Biol. 10 (4) (2016) 136–146.
[3] M. MashhadiMalek, F. Towhidkhah, S. Gharibzadeh, V. Daeichin, M.A. Ahmadi-Pajouh, Are rigidity and tremor two sides of the same coin in Parkinson's disease? Comput. Biol. Med. 38 (11–12) (2008) 1133–1139.
[4] P. Rack, H. Ross, The role of reflexes in the resting tremor of Parkinson's disease, Brain 109 (1) (1986) 115–141.
[5] R. Edwards, A. Beuter, L. Glass, Parkinsonian tremor and simplification in network dynamics, Bull. Math. Biol. 61 (1) (1999) 157–177.
[6] D. Terman, J.E. Rubin, A. Yew, C. Wilson, Activity patterns in a model for the subthalamopallidal network of the basal ganglia, J. Neurosci. 22 (7) (2002) 2963–2976.
[7] M. Haeri, Y. Sarbaz, S. Gharibzadeh, Modeling the Parkinson's tremor and its treatments, J. Theor. Biol. 236 (3) (2005) 311–322.
[8] P. Pakarian, S.M. Rayegani, S. Shahzadi, Effect of Vim thalamic DBS in Parkinson's disease on F wave duration, Neurosci. Lett. 367 (3) (2004) 323–326.
[9] Y. Naito, Y. Komatsu, I. Kanazawa, T. Nakanishi, F response abnormality in Parkinson's disease, Psychiatry Clin. Neurosci. 42 (4) (1988) 811–818.
[10] I. Milanov, Motoneuron activity in patients with different types of tremor, Electromyogr. Clin. Neurophysiol. 41 (8) (2001) 479–484.
[11] P. Le Cavorzin, G. Carrault, F. Chagneau, P. Rochcongar, H. Allain, A computer model of rigidity and related motor dysfunction in Parkinson's disease, Movement

disorders: official journal of the Movement Disorder Society 18 (11) (2003) 1257–1265.
[12] A. Gigante, et al., Action tremor in Parkinson's disease: frequency and relationship to motor and non-motor signs, Eur. J. Neurol. 22 (2) (2015) 223–228.
[13] D. Kübler, et al., Determining an efficient deep brain stimulation target in essential tremor-Cohort study and review of the literature, Parkinsonism Relat. Disord. 89 (2021) 54–62.
[14] Y. Sui, et al., Deep brain stimulation initiative: toward innovative technology, new disease indications, and approaches to current and future clinical challenges in neuromodulation therapy, Front. Neurol. 11 (2021) 1706.
[15] B. Faraji, M. Gheisarnejad, M. Yalsavar, M.-H. Khooban, An Adaptive ADRC Control for Parkinson's Patients Using Machine Learning, IEEE Sens. J. (2020).
[16] M. Khan, J. Paktiawal, R.J. Piper, A. Chari, M.M. Tisdall, "Intracranial neuromodulation with deep brain stimulation and responsive neurostimulation in children with drug-resistant epilepsy: a systematic review," *Journal of Neurosurgery*, Pediatrics vol. 1, no. aop (2021) 1–10.
[17] K. Rouhollahi, M.E. Andani, J.A. Marnanii, S.M. Karbassi, Rehabilitation of the Parkinson's tremor by using robust adaptive sliding mode controller: a simulation study, IET Syst. Biol. 13 (2) (2019) 92–99.
[18] B. Faraji, M. Gheisarnejad, Z. Esfahani, and M.-H. Khooban, "Smart Sensor Control for Rehabilitation in Parkinson's Patients," *IEEE Transactions on Emerging Topics in Computational Intelligence,* 2021.
[19] B. Faraji, Z. Esfahani, K. Rouhollahi, D. Khezri, Optimal Canceling of the Physiological Tremor for Rehabilitation in Parkinson's disease, Journal of Exercise Science and Medicine (JESM) 11 (2) (2019) 113–124.
[20] C. Liu, et al., Closed-loop control of tremor-predominant parkinsonian state based on parameter estimation, IEEE Trans. Neural Syst. Rehabil. Eng. 24 (10) (2016) 1109–1121.
[21] A. Horn, A.A. Kühn, Lead-DBS: a toolbox for deep brain stimulation electrode localizations and visualizations, Neuroimage 107 (2015) 127–135.
[22] M. Gheisarnejad, H. Farsizadeh, M.H. Khooban, A Novel Nonlinear Deep Reinforcement Learning Controller for DC–DC Power Buck Converters, IEEE Trans. Ind. Electron. 68 (8) (2020) 6849–6858.
[23] M. Gheisarnejad, H. Farsizadeh, M.-R. Tavana, M.H. Khooban, A Novel Deep Learning Controller for DC–DC Buck-Boost Converters in Wireless Power Transfer Feeding CPLs, IEEE Trans. Ind. Electron. 68 (7) (2020) 6379–6384.
[24] M.-B. Radac, R.-E. Precup, Data-based two-degree-of-freedom iterative control approach to constrained non-linear systems, IET Control Theory Appl. 9 (7) (2015) 1000–1010.
[25] Z. Yan, Y. Xu, Data-driven load frequency control for stochastic power systems: A deep reinforcement learning method with continuous action search, IEEE Trans. Power Syst. 34 (2) (2018) 1653–1656.
[26] J. Ling, Z. Feng, M. Ming, X. Xiao, Damping controller design for nanopositioners: A hybrid reference model matching and virtual reference feedback tuning approach, Int. J. Precis. Eng. Manuf. 19 (1) (2018) 13–22.
[27] M. Gheisarnejad, M.H. Khooban, An intelligent non-integer PID controller-based deep reinforcement learning: Implementation and experimental results, IEEE Trans. Ind. Electron. 68 (4) (2020) 3609–3618.
[28] B. Faraji, M. Gheisarnejad, K. Rouhollahi, Z. Esfahani, M.H. Khooban, Machine learning approach based on ultra-local model control for treating cancer pain, IEEE Sens. J. 21 (6) (2020) 8245–8252.
[29] B. Faraji, D. Khezri, Ultra-Local Model Control of Parkinson's Patients Based on Machine Learning, Journal of Advanced Sport Technology 5 (1) (2021) 1–16.
[30] T. MohammadRidha, et al., Model free iPID control for glycemia regulation of type-1 diabetes, IEEE Trans. Biomed. Eng. 65 (1) (2017) 199–206.
[31] A.V. Hill, First and last experiments in muscle mechanics, Cambridge University Press, 1970.
[32] P.J. Delwaide, Parkinsonian rigidity, Funct. Neurol. 16 (2) (2001) 147–156.
[33] H. Ahmed, I. Salgado, H. Ríos, Robust synchronization of master-slave chaotic systems using approximate model: An experimental study, ISA Trans. 73 (2018) 141–146.
[34] A.G. Haroun, Y.-Y. Li, A novel optimized hybrid fuzzy logic intelligent PID controller for an interconnected multi-area power system with physical constraints and boiler dynamics, ISA Trans. 71 (2017) 364–379.
[35] C. Wang, J. Wang, Y. Shen, X. Zhang, Autonomous navigation of UAVs in large-scale complex environments: A deep reinforcement learning approach, IEEE Trans. Veh. Technol. 68 (3) (2019) 2124–2136.
[36] M. Zhu, X. Wang, Y. Wang, Human-like autonomous car-following model with deep reinforcement learning, Transportation research part C: emerging technologies 97 (2018) 348–368.