

# Deep Learning Model Predictive Control for Deep Brain Stimulation in Parkinson’s Disease

Sebastian Steffen\*

Mark Cannon\*

**Abstract**— We present a nonlinear data-driven Model Predictive Control (MPC) algorithm for deep brain stimulation (DBS) for the treatment of Parkinson’s disease (PD). Although DBS is typically implemented in open-loop, closed-loop DBS (CLDBS) uses the amplitude of neural oscillations in specific frequency bands (e.g. beta 13-30 Hz) as a feedback signal, resulting in improved treatment outcomes with reduced side effects and slower rates of patient habituation to stimulation. To date, CLDBS has only been implemented *in vivo* with simple algorithms such as proportional, proportional-integral, and thresholded switching control. Our approach employs a multi-step predictor based on differences of input-convex neural networks to model the future evolution of beta oscillations. The use of a multi-step predictor enhances prediction accuracy over the optimization horizon and simplifies online computation. In tests using a simulated model of beta-band activity response and data from PD patients, we achieve reductions of more than 20% in both tracking error and control activity in comparison with existing CLDBS algorithms. The proposed control strategy provides a generalizable data-driven technique that can be applied to the treatment of PD and other diseases targeted by CLDBS, as well as to other neuromodulation techniques.

**Index Terms**— neuromodulation, data-driven control, model predictive control, nonlinear systems

## I. INTRODUCTION

Deep Brain Stimulation (DBS) is a treatment for various neurological and psychiatric disorders that involves the surgical implantation of electrodes into specific structures deep within the brain. DBS devices deliver pulses of electric current to disrupt pathological activity in the central nervous system. The technique is currently used to treat essential tremor, Parkinson’s disease (PD), and epilepsy [1]. It is also being trialed for treatment-resistant depression and obsessive-compulsive disorder, among other conditions [2], [3]. Typically, DBS operates in an open-loop mode with a fixed stimulation pattern of constant amplitude, frequency and pulse-width, which results in high stimulation levels compared to targeted stimulation algorithms [4]. Excessive stimulation can increase side-effects and lead to a more rapid decline in the efficacy of treatment due to habituation [5]. This has motivated extensive research into closed-loop DBS (CLDBS), leading to recent approval of the technique for PD patients [6], [7].

Typically, CLDBS algorithms typically modulate stimulation amplitude as a function of disease biomarkers measured by the implant [8]. Symptoms of PD are associated with

bursts in amplitude of so-called beta-band oscillations (13-30Hz) in population-level neural activity known as local field potentials (LFP) [9], [10]. To date, *in vivo* CLDBS has been limited to simple algorithms such as on/off switching control [11], [12], proportional or proportional-integral (PI) control [13], or dual-threshold control [14], [15].

Simulation-based studies allow a wider range of control strategies to be investigated. Some approaches incorporate additional information, such as muscle activity, to select between PI controllers tuned for different operating points [16]. Others employ different control objectives, e.g., minimizing the duration of periods of high pathological activity [17]. Another research direction is to use model-based feedback control laws. In [18] an LQR controller is designed to control the frequency of DBS pulses based on a model derived from input-output data, and in [19] this is augmented with nonlinear terms to improve model accuracy, and the controller contains additional terms to cancel the nonlinearity.

Several studies have proposed optimal predictive control schemes, although no consensus has emerged on the type of model to use. The majority of schemes fit a linear model to the dynamics of the relevant biomarker, for example, ARX models [20] or state-space models fit using subspace methods [21]. In our previous work [22], we proposed an augmented model comprising an online-identified linear model of biomarker activity and a second-order model of the response to stimulation based on averaged patient data. While linear models can provide a simpler formulation of the optimal control problem, they do not accurately capture the inherently nonlinear dynamics of neural systems [23]. Very few studies have investigated nonlinear predictive control. In [24], a Volterra series is used to represent the nonlinear dynamics of the patient response to DBS. To the best of our knowledge, no studies have considered the use of more expressive nonlinear models such as feedforward or recurrent neural networks in the context of CLDBS.

If the nonlinear model can be represented as a difference of convex (DC) functions, the online MPC optimization problem can be solved efficiently as a sequence of convex problems [25], [26]. These are derived via partial linearization of the system model around nominal predicted trajectories. The DC model representation provides tight bounds on linearization errors and the nominal trajectories are successively updated using the most recent solution estimate. For example, [27] uses input-convex neural networks (ICNN [28]) to apply robust MPC to a batch bioreactor. The resulting DC-MPC controller, like other robust nonlinear

This work was supported by the UKRI Engineering and Physical Sciences Research Council

\*Department of Engineering Science, University of Oxford, OX1 3PJ, UK. {sebastian.steffen,mark.cannon}@eng.ox.ac.uk

MPC approaches, requires the propagation of linearization errors over the prediction horizon to ensure satisfaction of constraints. As discussed in [29], a multi-step predictor model simplifies the construction of robust tubes bounding the future model state.

*Contribution:* This paper provides the following contributions to the state of the art in CLDBS.

- We formulate a nonlinear optimal control problem for CLDBS subject to input and output constraints, using a multi-step predictor model defined for each prediction step by the difference of a pair of ICNNs.
- We solve this problem using sequential convex programming with tight bounds on approximation errors.
- We construct the control law as a robust tube MPC strategy that explicitly accounts for linearization errors and uncertainty in the predictor model.
- The use of a multi-step predictor simplifies online computation by avoiding the need for recursive propagation of linearization errors over the prediction horizon.

The remainder of this paper is structured as follows. Section II outlines the problem formulation for CLDBS. Section III describes the proposed control law. Section IV describes numerical simulations and provides a discussion of their results. Finally, Section V provides concluding remarks and some perspectives on future work.

## II. PROBLEM FORMULATION

The envelope of the beta-band oscillations measured at the implant site in the subthalamic nucleus at time  $t$  is denoted  $y(t) \in \mathbb{R}^+$ . We assume that the dynamics of the biomarker  $y(t)$  are given by an unknown, nonlinear (and possibly time-varying) dynamical system of the form

$$\dot{y}(t) = f(y(t), u(t), t) \quad (1)$$

where the control input  $u(t)$  is the product of the applied stimulation amplitude in volts, pulse width in seconds and frequency in Hertz, and is constrained to lie in the range  $u \in [0, u_{\max}]$  which is determined for each patient by a clinician. Furthermore, the rate of change of input is also constrained, lying in the range  $\dot{u}(t) \in [-\dot{u}_{\max}, \dot{u}_{\max}]$ . We assume that  $f$  is a continuous, twice-differentiable function, and can therefore be represented with arbitrary accuracy by a difference of convex functions.

The goal of CLDBS is to suppress beta-band activity exceeding a pathological level, which we denote  $y_0$ , while minimizing the stimulation energy. This suggests a cost index for the optimal control law defined over a horizon of length  $T$  of the form:

$$\int_0^T \left( \phi([y(t) - y_0]_{\geq 0}) + Ru(t)^2 \right) dt \quad (2)$$

$$[y(t) - y_0]_{\geq 0} = \begin{cases} y(t) - y_0, & \text{if } y(t) \geq y_0 \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

where  $\phi : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$  is a monotonically non-decreasing function, and  $R$  is a positive control weighting.

## III. CONTROL LAW

### A. Difference of Convex Functions Neural Network Model

To derive an MPC strategy we construct a discrete time model of the system (1). For the predictor model in discrete time, we define a separate neural network for each of the  $N$  steps of the prediction horizon:

$$\begin{bmatrix} y_{k+1} \\ y_{k+2} \\ \vdots \\ y_{k+N} \end{bmatrix} = \begin{bmatrix} f_1(z_k, u_k) \\ f_2(z_k, u_{k:k+1}) \\ \vdots \\ f_N(z_k, u_{k:k+N-1}) \end{bmatrix} + \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix} \quad (4)$$

where  $k$  is the discrete time index for a given sampling interval and each function  $f_i$  is a difference of convex functions

$$f_i(z_k, u_{k:k+i-1}) = f_{i,1}(z_k, u_{k:k+i-1}) - f_{i,2}(z_k, u_{k:k+i-1}). \quad (5)$$

The arguments of  $f_i, f_{i,1}, f_{i,2}$  are  $z_k = [y_{\text{past},k} \ u_{\text{past},k}]^T$  which contains  $n_y$  past observations of the system output and  $n_u$  control inputs, i.e.  $y_{\text{past},k} = [y_k \ y_{k-1} \ \cdots \ y_{k-n_y+1}]$  and  $u_{\text{past},k} = [u_{k-1} \ u_{k-2} \ \cdots \ u_{k-n_u}]$ , and the sequence of control inputs from timestep  $k$  until timestep  $k+i-1$ , i.e.  $u_{k:k+i-1} = [u_k \ u_{k+1} \ \cdots \ u_{k+i-1}]^T$ , and the disturbance  $w_i \in \mathbb{W}_i = [w_{i,\min}, w_{i,\max}]$  accounts for modeling error.

The functions  $f_{i,1}, f_{i,2}$  are each specified by ICNNs with the structure proposed in [28], consisting of a series of fully connected hidden layers, with additional skip connections from the input to the hidden layers. The use of rectified linear units as activation functions, together with non-negative constraints on the dense weights between hidden layers ensures convexity with respect to the network inputs.

### B. DCNN Tube MPC

We propose a robust MPC law obtained by minimizing a discrete-time, convex approximations of the cost (2). Each convex approximation is obtained by linearizing the concave parts of the model (4) around a nominal trajectory  $(\bar{y}_k^0, \bar{u}_k^0)$ , where  $\bar{y}_k^0 = [y_{k+1}^0 \ \cdots \ y_{k+N}^0]$ ,  $\bar{u}_k^0 = [u_k^0 \ \cdots \ u_{k+N-1}^0]$ . We define perturbations,  $s_k = y_k - y_k^0$ , and  $v_k = u_k - u_k^0$ , and sets  $\mathbb{S}_{k+i} = [s_{k+i,\min}, s_{k+i,\max}]$  bounding the predicted deviation of  $y_{k+i}$  from  $y_{k+i}^0$  at time  $k+i$ . The DC property of  $f_i$  allows us to find tight bounds on the perturbations satisfying:

$$\begin{aligned} s_{k+i,\max} &\geq f_{i,1}(z_k, u_{k:k+i-1}) - f_{i,2}(z_k, u_{k:k+i-1}^0) \\ &\quad - f'_{i,2}(z_k, u_{k:k+i-1}^0)v_{k:k+i-1} + \max_{w \in \mathbb{W}_i} w - y_{k+i}^0 \end{aligned} \quad (6)$$

$$\begin{aligned} s_{k+i,\min} &\leq -f_{i,2}(z_k, u_{k:k+i-1}) + f_{i,1}(z_k, u_{k:k+i-1}^0) \\ &\quad + f'_{i,1}(z_k, u_{k:k+i-1}^0)v_{k:k+i-1} + \min_{w \in \mathbb{W}_i} w - y_{k+i}^0 \end{aligned} \quad (7)$$

where  $f'_{i,1}(z_k, u_{k:k+i-1}^0)$  and  $f'_{i,2}(z_k, u_{k:k+i-1}^0)$  denote the Jacobian matrices  $\partial f_{i,1}/\partial u$  and  $\partial f_{i,2}/\partial u$  evaluated along  $\bar{u}_k^0$ .

### C. Optimal MPC Problem

We define the worst-case MPC cost

$$J(z_k, \vec{u}_k, \vec{S}_k) = \sum_{i=0}^N \max_{s_{k+i} \in \vec{S}_{k+i}} \left( Q[y_{k+i}^0 + s_{k+i} - \beta_0]_{\geq 0}^2 + R u_{k+i}^2 \right) \quad (8)$$

where  $Q$  and  $R$  are positive, scalar weights, and  $[\cdot]_{\geq 0}$  denotes projection onto the positive orthant so that the term  $Q[y_{k+i}^0 + s_{k+i} - \beta_0]_{\geq 0}^2$  only penalizes the predicted values of  $y$  exceeding the threshold  $\beta_0$ . We denote  $\vec{u}_k^*$  as the solution of the convex program:

$$\text{minimize } J(z_k, \vec{u}_k, \vec{S}_k) \quad (9)$$

$$\vec{u}_k, \vec{S}_k$$

subject to (6), (7) and

$$\begin{aligned} y_{k+i}^0 + S_{k+i} &\subseteq \mathbb{Y} = [y_{\min}, y_{\max}] \\ u_{k+i} &\in \mathbb{U} = [u_{\min}, u_{\max}] \\ \Delta u_{k+i} &\in \Delta \mathbb{U} = [-\Delta u_{\max}, \Delta u_{\max}] \end{aligned}$$

for all  $i \in \{0, \dots, N-1\}$ . Here  $\mathbb{Y}$ ,  $\mathbb{U}$  and  $\Delta \mathbb{U}$  are the output, control input, and input rate constraint sets, with  $\Delta u_k = u_k - u_{k-1}$ . The solution of this optimal control problem is used to update  $\vec{u}_k^0$  and hence compute the nominal predicted trajectory  $\vec{y}_k^0$  using (4) with  $w_i = 0$  for all  $i$ . We iteratively update this solution until convergence (indicated by the change in the optimal cost falling below a threshold  $\Delta J_{\min}$ ), or until the maximum number of iterations (*maxiters*) is reached, as outlined in Algorithm 1.

---

#### Algorithm 1 Multi-step DCNN TMPC

---

**Require:**  $z_k$ , feasible  $\vec{y}_k^0$  and  $\vec{u}_k^0$

- 1:  $j \leftarrow 1$ ,  $\Delta J \leftarrow 10^6$ ,  $J_0 \leftarrow 10^6$
- 2: **while**  $j \leq \text{maxiters}$  **and**  $\Delta J > \Delta J_{\min}$  **do**
- 3: Evaluate the Jacobian matrices  $f'_{i,1}$  and  $f'_{i,2}$  using  $\vec{u}_k^0$
- 4: Solve problem (9) for  $\vec{u}_{k,j}^*$ , given  $\vec{y}_k^0$  and  $\vec{u}_k^0$
- 5:  $J_j \leftarrow J(z_k, \vec{u}_{k,j}^*)$ ,  $\Delta J \leftarrow J_j - J_{j-1}$   
 $\vec{u}_k^0 \leftarrow \vec{u}_{k,j}^*$ ,  $y_{k+i}^0 \leftarrow f_i(z_k, u_{k:k+i-1}^0)$ ,  $i = 1, \dots, N$
- 6:  $j \leftarrow j + 1$
- 7: **end while**
- 8: **return**  $\vec{u}_k^*$ ,  $\vec{y}_k^0$ ,  $\vec{u}_k^0$

---

## IV. NUMERICAL SIMULATIONS

We evaluate the performance of the proposed control scheme using LFP data from four individual Parkinsonian patients who underwent DBS surgery at either King's College Hospital or St George's Hospital in London. The LFP data was gathered while the patients were in a resting state, with stimulation switched off, for periods ranging from 15 to 30 minutes. The noisy LFP data was band-pass filtered between 18 and 24 Hz using a 6th order causal Butterworth filter, and the envelope was extracted via continuous wavelet transform. As in [22], we use here a synthetic model of the patients' beta-band activity in response to applied stimulation. The envelope of the beta-band oscillations is related to the

nominal activity  $y_\beta$  (i.e. the brain's activity in absence of any stimulation) and the DBS attenuation effect  $\eta(t)$  as follows,

$$y(t) = y_\beta(t) \cdot e^{-\eta(u(t))} \quad (10)$$

The stimulation response  $\eta(t)$  is represented by a second-order continuous-time system,

$$\begin{aligned} \dot{x}_c(t) &= \begin{bmatrix} -1/\tau_1(t) & 0 \\ g(t)/\tau_2(t) & -1/\tau_2(t) \end{bmatrix} x_c(t) + \begin{bmatrix} g(t)/\tau_1(t) \\ 0 \end{bmatrix} u(t) \\ \eta(t) &= [0 \quad 1] x_c(t) \end{aligned} \quad (11)$$

where the parameters  $\tau_1(t)$ ,  $\tau_2(t)$  and  $g(t)$  differ across patients, and also vary over time due to short-term changes in the patient's activity and long-term changes due to disease progression. The average values for these parameters are

$$\bar{g} = 62.11, \quad \bar{\tau}_1 = 0.05, \quad \bar{\tau}_2 = 0.25.$$

### A. Stimulation response and model training

Simulations use a discrete-time representation of the synthetic beta response model (11) employing a zero-order hold with sample rate 50 Hz. The parameters  $\tau_1(t)$ ,  $\tau_2(t)$  and  $g(t)$  of the model (11) vary in a random walk, with steps at each sampling instant of no more than 2.5% of their nominal value ( $\bar{\tau}_1$ ,  $\bar{\tau}_2$  or  $\bar{g}$ ) and with the constraint that the total variation does not exceed 40% of the nominal value.

The ICNNs were implemented using Keras [30] and trained on synthetically modulated LFP data sampled at 50 Hz. Training inputs were constructed using a pseudo-random binary sequence alternating between  $-\Delta u_{\max}$  and  $\Delta u_{\max}$  to define the incremental signal  $\Delta u_k = u_k - u_{k-1}$  subject to  $u_k \in [0, u_{\max}]$ . Networks were initially trained on  $10^5$  samples of synthetically modulated trajectories of LFP data taken from three patients (to create the 'pre-trained' model), then further trained on  $3.2 \times 10^4$  samples from the fourth patient (to create the 'refined' model), with  $10^4$  samples kept as the test set and with an offset of 3000 samples between test and training sets. In all experiments, we compare the performance of the pre-trained model and the refined model to test the ability of the model to generalize to beta activity of an unseen patient. 100 epochs of training 5 models on an Apple M1 Pro CPU took 9 minutes, 20 seconds.

### B. Multi-step Prediction Accuracy

We first compare: (i) the multi-step predictions of the refined model, (ii) the recursive application of the single-step ahead predictor ( $f_1(z_k, u_k)$  in (4)), and (iii) a linear model of beta oscillations (which is used in the linear MPC algorithm described in Section IV-C). For these tests, the predictors were trained and tested using patient activity with no stimulation effect. As expected, prediction errors increase as the prediction horizon grows. Qualitatively, the multi-step predictor is less smooth than either of the recursive predictors (Figure 2) but is on average closer to the true beta activity. Figure 3 shows that the multi-step predictor achieves the smallest mean absolute error across all prediction steps, but the maximum error is lower for the recursive predictor for 3 or more prediction steps.

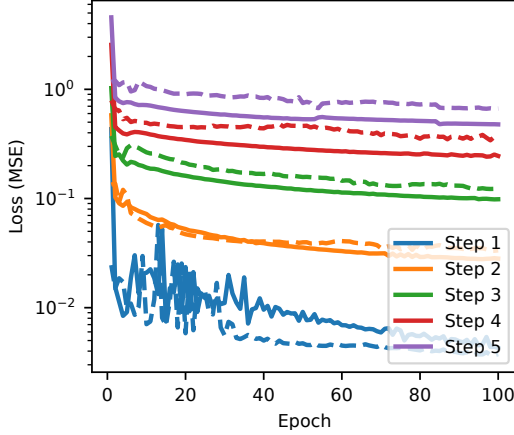


Fig. 1. Training (solid) and validation (dashed) loss against training epoch for the multi-step predictor model over 1-5 steps

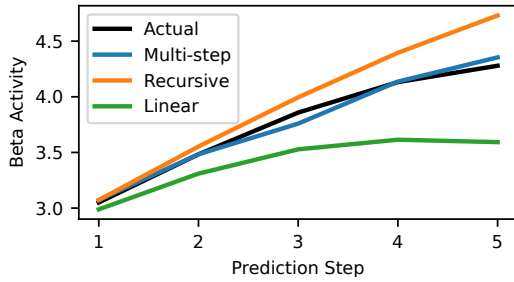


Fig. 2. Example of predicted trajectories over a 5-step horizon for the multi-step DCNN, and recursive predictions with the single-step DCNN and linear ARI model, compared to the true beta activity

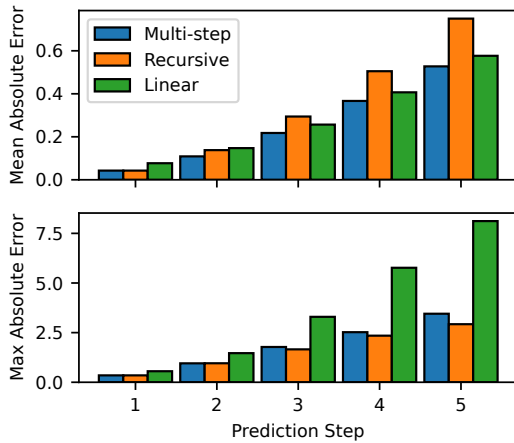


Fig. 3. Mean and maximum absolute errors for  $i = 1, \dots, 5$ -step ahead predictions for the multi-step DCNN, for recursive predictions using a single-step DCNN, and for the predictions of the linear ARI model

### C. Comparison with Alternative Control Strategies

This section compares the performance of DCNN TMPC with two control algorithms that have been tested *in vivo*, and with the linear MPC strategy proposed in [22]. We consider 50 simulations, each consisting of 15 seconds of patient data selected randomly from the test set, with the parameters of (11) varying stochastically as described in Section IV-A.

The on-off controller,  $u_{\text{on-off},k}$ , increases the level of stimulation by the maximum increment  $\Delta u_{\text{max}}$ , up to the maximum value  $u_{\text{max}}$  whenever the measured beta activity exceeds the threshold  $\beta_0$ , and reduces stimulation otherwise

$$u_{\text{on-off},k} = \begin{cases} \min(u_{\text{on-off},k-1} + \Delta u_{\text{max}}, u_{\text{max}}) & \text{if } y_k > \beta_0 \\ \max(u_{\text{on-off},k-1} - \Delta u_{\text{max}}, 0) & \text{otherwise.} \end{cases} \quad (12)$$

The PI controller is implemented in difference form,  $u_{\text{PI},k} = u_{\text{PI},k-1} + \Delta u_{\text{PI},k}$ , using the error signal  $e_k = [y(kT_s) - \beta_0]_{\geq 0}$ ,

$$\Delta u_{\text{PI},k} = K_P \Delta e_k + K_I T_s e_k \quad (13)$$

where  $K_P$  and  $K_I$  are the controller gains, with  $\Delta u$  and  $u$  limited (via saturation) to  $[-\Delta u_{\text{max}}, \Delta u_{\text{max}}]$  and  $[0, u_{\text{max}}]$ .

The linear MPC controller is described in detail in [22]; here we describe only its main features. The approach uses a linearizing transformation of (10),  $\xi_k = \ln(y_k)$ , such that the effect of stimulation appears additively in the model,  $\xi_k = \xi_{\beta,k} - \xi_{\eta,k}$ , and a linear ARI model of nominal beta activity

$$\Delta \xi_{\beta,k} = \sum_{i=1}^{n_{\beta}} \theta_i \Delta \xi_{\beta,k-i} \quad (14)$$

where  $\Delta \xi_{\beta,k} = \xi_{\beta,k} - \xi_{\beta,k-1}$ . The parameters  $\theta_1, \dots, \theta_{n_{\beta}}$  are identified from the patient data using a least squares approach over an identification period (during which there is no stimulation) immediately prior to initiating closed-loop control. The controller uses an augmented linear model

$$\begin{aligned} x_{k+1} &= A(\theta)x_k + Bu_k \\ A &= \begin{bmatrix} A_{\eta} & \mathbf{0}_{2 \times (n_{\beta}+1)} \\ \mathbf{0}_{(n_{\beta}+1) \times 2} & A_{\beta}(\theta) \end{bmatrix} \\ B &= \begin{bmatrix} B_{\eta} \\ \mathbf{0}_{(n_{\beta}+1) \times 1} \end{bmatrix} \\ \xi_k &= \begin{bmatrix} 0 & 1 & -1 & \mathbf{0}_{1 \times n_{\beta}} \end{bmatrix} x_k \end{aligned} \quad (15)$$

where the matrices  $A_{\eta}$  and  $B_{\eta}$  are computed by discretizing the model (11) with the average parameter values and a zero-order hold, and  $A_{\beta}(\theta)$  is the linear nominal activity model (14) in state-space form. The controller solves the optimal control input for a quadratic cost index

$$\underset{u}{\operatorname{argmin}} \sum_{i=0}^{N-1} \left( [\xi_{k+i} - \xi_0]_{\geq 0}^2 + Ru_{k+i}^2 \right) \quad (16)$$

subject to (15),  $\Delta u_{k+i} \in \Delta \mathbb{U}$  and  $u_{k+i} \in \mathbb{U}$  for all  $i \in \{0, \dots, N-1\}$ .

The DCNN TMPC algorithm was implemented in Python using the `cvxpy` library [31] to interface with the MOSEK solver. The maximum output constraint  $y_{\text{max}}$  was chosen

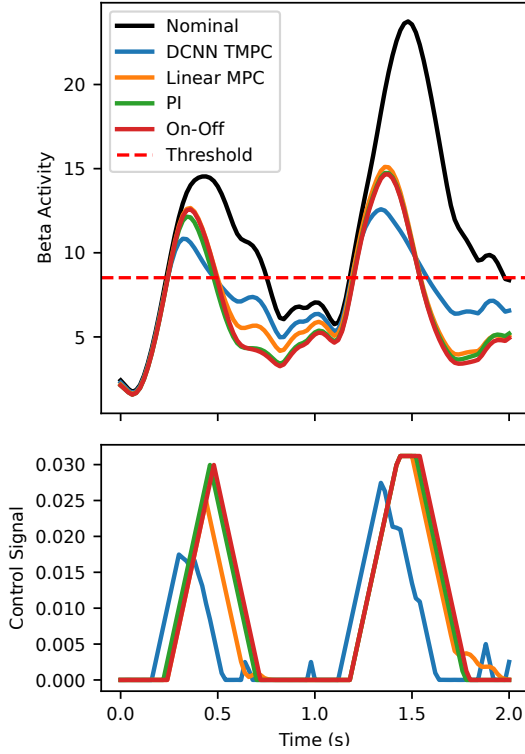


Fig. 4. Comparison of trajectories and control input sequences for DCNN TMPC (using the refined model), linear MPC, PI and on-off controllers. Here ‘Nominal’ indicates the raw beta activity with no applied stimulation

as the 95<sup>th</sup> percentile of the patient’s beta activity, while  $y_{min}$  was chosen as the minimum recorded value. We used a prediction horizon of  $N = 5$  steps, and the disturbance set  $\mathbb{W}_i$  for each  $i \in \{1, \dots, 5\}$  was chosen as the 80<sup>th</sup> percentile absolute prediction error during training. We note that this constraint cannot guarantee robust output constraint satisfaction. However, this is not problematic in practice because the safety of the DBS system is guaranteed by appropriately chosen constraints on the input  $u$  and rate of change  $\Delta u$ , so violations of output constraints can be handled by progressive constraint softening until the problem once again become feasible.

Figure 5 shows that DCNN TMPC provides significant improvements in performance compared to the linear MPC, PI and on-off controllers. With the refined model, DCNN TMPC beta suppression error is on average 30-50% higher for the alternative controllers. In addition, DCNN TMPC with only the pre-trained model outperforms linear MPC by at least 10% on average, and outperforms PI and on-off controllers by at least 20%. There is a smaller variation in the applied control input, however in all simulations, DCNN TMPC is more than 5% more energy-efficient than linear MPC, and 20% better than PI and on-off control. This highlights the benefits of using a more expressive nonlinear model for predicting the future system behavior. Moreover, the control input sequences in Fig. 4 show that DCNN TMPC responds more quickly than the alternative controllers to bursts in beta activity and reduces stimulation faster in

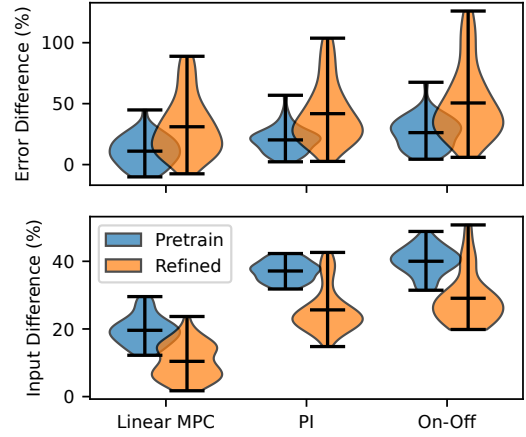


Fig. 5. Violin plot of percentage improvement in beta suppression error and control input for the DCNN TMPC algorithm with pretrained and refined model, relative to linear MPC, PI and on-off controllers for the 50 simulations. Bars show the mean, min, and max of the data

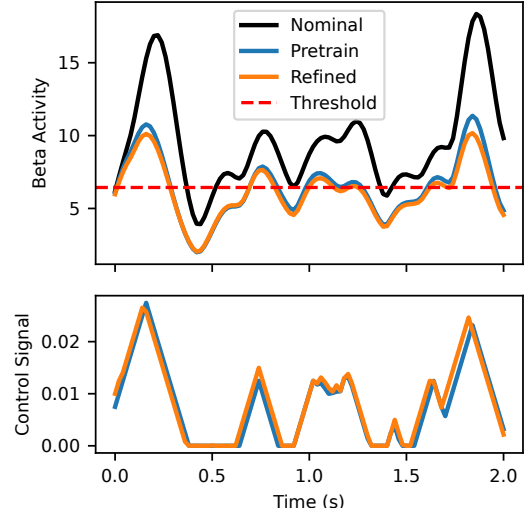


Fig. 6. Comparison of trajectories and control sequence for DCNN TMPC using a model with pre-training only, versus the refined model

response to decreases in beta activity. The beta activity plots also show that the proposed controller uses less unnecessary stimulation; this can be seen from the fact that the DCNN TMPC modulated trajectory is much closer to the nominal activity when it is below the threshold.

Comparing performance with pre-trained and refined models reveals significant robustness to differences in beta activity across different patients. Figure 5 shows that even with a model that was not trained on data from the test patient, DCNN TMPC is able to perform significantly better than the alternative controllers. Remarkably this observation even applies when DCNN TMPC with a pre-trained model is compared with linear MPC, which uses a model trained on data from the test patient. The comparison of the trajectories of DCNN TMPC with pre-training and refined models in Fig. 6 shows very similar trajectories, with the refined model applying slightly more control action, and generally acting faster, which results in better suppression of pathological beta

activity. Generally it appears that the model with pre-training alone results in a less aggressive controller.

## V. CONCLUDING REMARKS AND FURTHER WORK

This paper demonstrates that DCNN TMPC with a multi-step predictor outperforms existing control algorithms for CLDBS in simulations using actual patient data. We demonstrate that the multi-step NN predictor performs better than recursive predictions, and that the neural network-based controller generalizes to differences in beta activity across different patients.

We believe our approach provides a readily generalizable framework for CLDBS, and indeed many other closed-loop neuromodulation techniques, as it makes few assumptions about the dynamic model underlying the response of the relevant bio-marker. This approach will remain useful even if a different biomarker (or set of biomarkers) is discovered to be a better indicator of disease state.

A limitation of the multi-step predictor approach is that it is unable to ensure recursive feasibility since there is no guarantee that the errors in multi-step predictions will be consistent for predictions made at different times. We aim to investigate methods of bounding prediction errors such that we can develop either robust or stochastic feasibility certificates. In addition, we plan to validate controller performance in biophysical models and *in vivo* with PD patients.

## REFERENCES

- [1] C. Sarica, C. Conner, K. Yamamoto, A. Yang, J. Germann, M. Lannon, *et al.*, “Trends and disparities in deep brain stimulation utilization in the United States: a nationwide inpatient sample analysis from 1993 to 2017,” *Lancet Regional Health. Americas*, p. 100599, 2023.
- [2] S. Groppa, G. Gonzalez-Escamilla, G. Tinkhauser, H. I. Baqapuri, B. Sajonz, C. Wiest, *et al.*, “Perspectives of implementation of closed-loop deep brain stimulation: From neurological to psychiatric disorders,” *Stereotactic and Functional Neurosurgery*, vol. 102, pp. 40–54, 12 2023.
- [3] K. Sellers, J. Cohen, A. Khambhati, M. Fan, A. Lee, E. Chang, *et al.*, “Closed-loop neurostimulation for the treatment of psychiatric disorders,” *Neuropsychopharmacol.*, vol. 49, p. 163–178, 2024.
- [4] S. J. Schiff, “Towards model-based control of Parkinson’s disease,” *Phil. Trans. R. Soc. A*, vol. 368, pp. 2269–2308, 2010.
- [5] J. Fleming, E. Dunn, and M. Lowery, “Simulation of closed-loop deep brain stimulation control schemes for suppression of pathological beta oscillations in Parkinson’s disease,” *Front. Neurosci.*, vol. 14, no. 166, 2020.
- [6] Medtronic plc, “Medtronic earns U.S. FDA approval for the world’s first adaptive deep brain stimulation system for people with Parkinson’s.” <https://news.medtronic.com/2025-02-24-Medtronic-earns-U-S-FDA-approval-for-the-worlds-first-Adaptive-deep-brain-stimulation-system-for-people-with-Parkinsons>, 2025. Last accessed 24 March 2025.
- [7] U.S. Food and D. Administration, “FDA roundup: February 28, 2025.” <https://www.fda.gov/news-events/press-announcements/fda-roundup-february-28-2025>, 2025. Last accessed 24 March 2025.
- [8] P. Acharyya, K. W. Daley, J. W. Choi, K. B. Wilkins, S. Karjagi, C. Cui, *et al.*, “Closing the loop in DBS: A data-driven approach,” *Parkinsonism & Related Disorders*, p. 107348, 2025.
- [9] P. Silberstein, A. Oliviero, V. Di Lazzaro, A. Insola, P. Mazzone, and P. Brown, “Oscillatory pallidal local field potential activity inversely correlates with limb dyskinesias in Parkinson’s disease,” *Exp. Neurol.*, vol. 194, pp. 523–529, 2005.
- [10] G. Tinkhauser, F. Torrecillos, A. Pogosyan, A. Mostofi, M. Bange, P. Fischer, *et al.*, “The cumulative effect of transient synchrony states on motor performance in Parkinson’s disease,” *J. Neurosci.*, vol. 40, pp. 1571–1580, 2020.
- [11] S. He, F. Baig, A. Merla, F. Torrecillos, A. Perera, C. Wiest, *et al.*, “Beta-triggered adaptive deep brain stimulation during reaching movement in Parkinson’s disease,” *Brain*, vol. 146, pp. 5015–5030, 07 2023.
- [12] M. Arlotti, S. Marceglia, G. Foffani, J. Volkmann, A. M. Lozano, E. Moro, *et al.*, “Eight-hours adaptive deep brain stimulation in patients with Parkinson disease,” *Neurology*, vol. 90, pp. e971–e976, 2018.
- [13] S. L. Schmidt, A. H. Chowdhury, K. T. Mitchell, J. J. Peters, Q. Gao, H.-J. Lee, *et al.*, “At home adaptive dual target deep brain stimulation in Parkinson’s disease with proportional control,” *Brain*, vol. 147, no. 3, pp. 911–922, 2023.
- [14] A. Velisar, J. Syrkin-Nikolau, Z. Blumenfeld, M. Trager, M. F. Afzal, V. Prabhakar, *et al.*, “Dual threshold neural closed loop deep brain stimulation in Parkinson disease patients,” *Brain Stimulation*, vol. 12, pp. 868–876, 2019.
- [15] S. Stanslaski, R. L. S. Summers, L. Tonder, Y. Tan, M. Case, R. S. Raikie, *et al.*, “Sensing data and methodology from the adaptive DBS algorithm for personalized therapy in Parkinson’s disease (ADAPT-PD) clinical trial,” *npj Parkinsons Disease*, vol. 10, p. 174, Sep 2024.
- [16] J. Fleming, S. Senneff, and M. Lowery, “Multivariable closed-loop control of deep brain stimulation for Parkinson’s disease,” *Journal of Neural Engineering*, vol. 20(5):10.1088/1741-2552/acfbfa, 2023.
- [17] M. Petrucci, R. Anderson, J. O’Day, K. Y.M., J. A. Herron, and H. Bronte-Stewart, “A closed-loop deep brain stimulation approach for mitigating burst durations in people with Parkinson’s disease,” in *Annual International Conference of the IEEE Engineering in Medicine and Biology Society.*, pp. 3617–3620, 2020.
- [18] Y. Yang, A. T. Connolly, and M. M. Shanechi, “A control-theoretic system identification framework and a real-time closed-loop clinical simulation testbed for electrical brain stimulation,” *Journal of Neural Engineering*, vol. 15(6):10.1088/1741-2552/aad1a8, p. 066007, 2018.
- [19] H. Fang, S. A. Berman, Y. Wang, and Y. Yang, “Robust adaptive deep brain stimulation control of in-silico non-stationary Parkinsonian neural oscillatory dynamics,” *Journal of Neural Engineering*, vol. 21(3):10.1088/1741-2552/ad5406, 2024.
- [20] A. Haddock, A. Velisar, J. Herron, H. Bronte-Stewart, and H. Chizeck, “Model predictive control of deep brain stimulation for Parkinsonian tremor,” in *8th International IEEE EMBS Conference on Neural Engineering*, (Shanghai, China), pp. 358–362, 2017.
- [21] M. Ahmadipour, M. Barkhordari-Yazdi, and S. Seydnejad, “Subspace-based predictive control of Parkinson’s disease: A model-based study,” *Neural Networks*, vol. 142, pp. 680–689, 2021.
- [22] S. Steffen, M. Cannon, H. Tan, and J. Debarros, “Model predictive control for closed-loop deep brain stimulation,” in *IEEE 63rd Conference on Decision and Control (CDC)*, 2024.
- [23] J. Rubin and D. Terman, “High frequency stimulation of the subthalamic nucleus eliminates pathological thalamic rhythmicity in a computational model,” *J. Comput. Neurosci.*, vol. 16, pp. 211–235, 2004.
- [24] F. Su, J. Wang, S. Niu, H. Li, B. Deng, C. Liu, *et al.*, “Nonlinear predictive control for adaptive adjustments of deep brain stimulation parameters in basal ganglia–thalamic network,” *Neural Networks*, vol. 98, pp. 283–295, 2018.
- [25] M. Doff-Sotta and M. Cannon, “Difference of convex functions in robust tube nonlinear MPC,” *2022 IEEE 61st Conference on Decision and Control (CDC)*, pp. 3044–3050, 2022.
- [26] Y. Lishkova and M. Cannon, “A successive convexification approach for robust receding horizon control,” *IEEE Transactions on Automatic Control (to appear)*, 2025.
- [27] N. Krausch, M. Doff-Sotta, M. Cannon, P. Neubauer, and M. N. Cruz-Bourmazou, “Deep learning adaptive model predictive control of fed-batch cultivations,” <https://arxiv.org/abs/2502.01488>, 2025.
- [28] B. Amos, L. Xu, and J. Z. Kolter, “Input convex neural networks,” in *International Conference on Machine Learning*, pp. 146–155, 2017.
- [29] J. Köhler, K. P. Wabersich, J. Berberich, and M. N. Zeilinger, “State space models vs. multi-step predictors in predictive control: Are state space models complicating safe data-driven designs?,” in *2022 IEEE 61st Conference on Decision and Control (CDC)*, pp. 491–498, 2022.
- [30] F. Chollet *et al.*, “Keras.” <https://keras.io>, 2015.
- [31] S. Diamond and S. Boyd, “CVXPY: A Python-embedded modeling language for convex optimization,” *Journal of Machine Learning Research*, vol. 17, no. 83, pp. 1–5, 2016.