Random Batch Method (RBM) for Converted-Dominated Power System Models

Control, Inverse Problems and Machine Learning

Carlos Vázquez Monzón

January 16, 2024

CHAIR OF COMPUTATIONAL MATHEMATICS

instituto de matemáticas universidad de sevilla

MINISTERIO DE CIENCIA, INNOVACIÓN Y UNIVERSIDADES

Introduction

- Stability and control aspects of converter-dominated power systems for integrating renewable energies.
- Implementation of the Random Batch Method (RBM) to the converter-dominated power systems model.

LQ optimal control problem

We consider the classical (finite-dimensional) LQ optimal control problem in which we want to find the control $u^*(t)$ that minimizes

$$J(u) = \frac{1}{2}\int_0^T \Big((x(t) - x_d(t))^T Q(x(t) - x_d(t)) + u(t)^T R u(t)\Big) \tag{1}$$

subject to the dynamics

$$\dot{x}(t) = Ax(t) + Bu(t), \; x(0) = x_0, \tag{2}$$

with x_d being the reference trajectory and Q and R the weighting matrices.

Random Batch Method (RBM) Implementation

- RBM involves a multi-step process for efficiently reducing the model order while retaining dynamic characteristics.
- Crucial for handling the complexity of converter-dominated power systems, particularly evident in its application to the electrical model.

The proposed randomized time-splitting method

Step 1. Decompose the matrix A into M submatrices A_m:

$$A = \sum_{m=1}^{M} A_m \tag{3}$$

It's preferable that each A_m is dissipative, i.e., $< x, A_m x > \leq 0$ $\forall x, m \in 1, 2..., M$.
We will choose $M = 3$.

Random Batch Method

Batch: array of size that of the number of the discretization points

Batch randomly chosen: $[1, 2, 1, 2, 2, 3, 1...]$

Matrix randomly chosen: $[A_1, A_2, A_1, A_2, A_2, A_3, A_1...]$

```
def calculate_states_random(Ah, b, tgrid, T, ntrials, M, K, X0):
    def matrix(t):
        # Define the matrix A as a function of time
        i = np.argmin(np.abs(tgrid - t))
        m = batches[i]
        print(t)
        return Ah[m-1]

    def system(t, x):
        # Evaluate A(t) at the current time t
        A_t = matrix(t)
        # Define the system of ODEs: x' = A(t) * x + b
        return np.dot(A_t, x) + b

    for j in range(0,1):
        x = []
        for _ in range(0,ntrials):
            batches = generate_batch(1, M, K)
            sol = solve_ivp(fun=lambda t, x: system(t, x), t_span=(0,T), y0=X0, method='Radau', t_eval=tgrid)
            x.append(sol.y[j, :])
        # Plot the solution
        plot_solutions_random(x,tgrid, j, ntrials)
```

The proposed randomized time-splitting method

Step 2. For each the 2^M subsets of $\{1,2,\ldots, M\}$, which we call $\{S_1, S_2, \ldots, S_{2^M}\}$, assign probabilities $p_1, p_2, \ldots, p_{2^M}$ to be chosen, such as

$$\sum_{l=1}^{2^M} p_l = 1 \tag{4}$$

and

$$\pi_m = \sum_{l \in L_m} p_l > 0, \tag{5}$$
$$L_m = \{l \in \{1, 2, \ldots, 2^M\} / m \in S_l\},$$

for all $m \in \{1, 2, \ldots, M\}$.

The proposed randomized time-splitting method

Step 3. Divide the considered time interval $[0, T]$ into K subintervals $[t_{k-1}, t_k)$, $k \in \{1, ..., K\}$:

$$0 = t_0 < t_1 < ... < t_{K-1} < t_K = T \tag{6}$$

and choose an index ω_k according to the probability distribution of the assigned probabilities $p_1, p_2, ..., p_{2^M}$ in each subinterval independently. Store the selected indices as

$$\omega := (\omega_1, \omega_2, ..., \omega_K). \tag{7}$$

The proposed randomized time-splitting method

Step 4. For the selected ω, we define a matrix $A_h(\omega, t)$

$$A_h(\omega, t) = \sum_{m \in S_{\omega_k}} \frac{A_m}{\pi_m}, \tag{8}$$

for $t \in [t_{k-1}, t_k)$. It can be easily proven that $E(A_h(\omega, t)) = A$.

The proposed randomized time-splitting method

Step 5. The matrix A is replaced by $A_h(\omega, t)$ in (1) and (2). Now we look to compute the solution $x_h(\omega, t)$ of the dynamics

$$\dot{x}_h(\omega, t) = A_h(\omega, t)x_h(\omega, t) + Bu(t), \;\; x_h(\omega, 0) = x_0 \tag{9}$$

for a given control $u(t)$.

Example. 1D heat equation with no controls

$$\frac{\partial y(t,\xi)}{\partial t} = \frac{\partial^2 y(t,\xi)}{\partial x^2},\ \xi \in [-L, L] \tag{10}$$

$$\frac{\partial y(t,-L)}{\partial t} = \frac{\partial y(t,L)}{\partial t} = 0, \tag{11}$$

$$y(0,\xi) = \exp(-\xi^2) + \xi^2 \exp(-L^2). \tag{12}$$

Crank Nicholson scheme for the heat equation

$$A = \frac{1}{\Delta\xi^2}\begin{bmatrix} -2 & 2 & 0 & \dots & 0 \\ 1 & -2 & 1 & \dots & 0 \\ 0 & 1 & -2 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 1 \\ 0 & 0 & \dots & 2 & -2 \end{bmatrix}$$

Previous decomposition by blocks

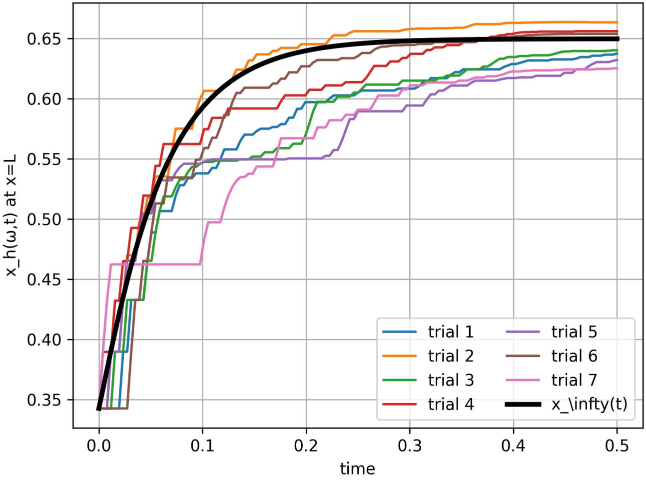


Figure: RBM for the 1D heat equation with no controls for 7 realizations

Electrical model

$$\begin{cases} \frac{di_g}{dt} & = \omega_b \left(\frac{1}{L_g}(v_0 - R_g i_g - v_g) - j\omega_g i_g \right) \\ \frac{di_i}{dt} & = \omega_b \left(\frac{1}{L_f}(mV_{dc} - R_f i_i - v_0) - j\omega_g i_i \right) \\ \frac{dv_0}{dt} & = \omega_b \left(\frac{1}{C_f}(i_i - i_g) - j\omega_g v_0 \right) \\ \frac{d(M_f i_f)}{dt} & = \frac{1}{K} \left[Q_{ref} - Q + K_q(\hat{v}_{ref} - \hat{v}_0^c) \right] \\ \frac{d\omega_{sv}}{dt} & = \frac{1}{T_a} \left[\frac{P_m}{\omega_{sv}} - \frac{P_e}{\omega_{sv}} - K_D(\omega_{sv} - \omega_{ref}) \right] \\ \frac{d\delta\theta_{sv}}{dt} & = \omega_b(\omega_{sv} - \omega_g), \end{cases} \tag{13}$$

being $P_m = P_{ref} + K_\omega(\omega_{ref} - \omega_{sv})$. v_0, i_g, v_g and i_i have two components: one in the d-axis (real part) and the other in the q-axis (imaginary part). We can therefore convert the complex ODE system into a real one, by introducing the variables $v_0^q, v_0^d, i_g^d, i_g^q$, etc.

Applying the RBM to the electrical model

States x: $i_g^d, i_g^q, i_i^d, i_i^q, v_0^d, v_0^q, M_f i_f, \omega_{sv}, \delta\theta_{sv}$ (9)
Controls u: $V_g^d, V_g^q, P_{ref}, Q_{ref}, \omega_{ref}, \hat{v}_{ref}$ (6). The controls are constant.

Linealized model

$A = [\nabla_x f(x,u)]_{x=x_0}$, $B = [\nabla_u f(x,u)]_{x=x_0}$, being x_0 a steady-state point.

$$A = \begin{pmatrix} -15.45 & 57.17 & 0 & 0 & 2128.99 & 0 & 0 & 0 \\ -1726.44 & -15.45 & 0 & 0 & 0 & 2078.63 & 0 & 0 \\ 0 & 0 & -11.82 & 57.17 & -5322.49 & 0 & 5272.42 & -266.42 \\ 0 & 0 & -1726.44 & -11.82 & 0 & -5196.58 & 6100.30 & 6953.81 \\ -3881.86 & 0 & 3881.86 & 0 & 0 & 55.82 & 0 & 0 \\ 0 & -3975.91 & 0 & 3975.91 & -1768.27 & 0 & 0 & 0 \\ 0 & 0 & -0.06 & -0.00 & -0.06 & -0.00 & -41.77 & 6.55e-19 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1308.99 & 0 \\ 0 & 0 & -0.00 & 0.00 & 0.00 & -0.00 & 0 & -3.16e-20 \end{pmatrix}$$

Linealized model

$$B = \begin{pmatrix} -3.78 & 0 & 0 & 0 & 0 & 0 \\ 0 & -20.79 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4.55e-8 & 0 & 0.13 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 7.18e-10 & 0 & 0 \end{pmatrix}$$

$$u = [\sqrt{2} \cdot 398.0, 0, 1375000.0, 0, 100.0\pi, 563.0]$$

Applying the RBM to the electrical model

Eigenvalues of the state matrix A: $[-6.42 + 5692.42j, -6.42 - 5692.42j, -6.42 + 5064.10j, -6.42 - 5064.10j, -14.42 + 314.16j, -14.42 - 314.16j, -39.44, -23.64, -4.65e - 03]$. All with negative real part.

$$A = Q\Lambda Q^{-1},\ sp(A) = \{\lambda_1, \lambda_2, ..., \lambda_9\}.$$

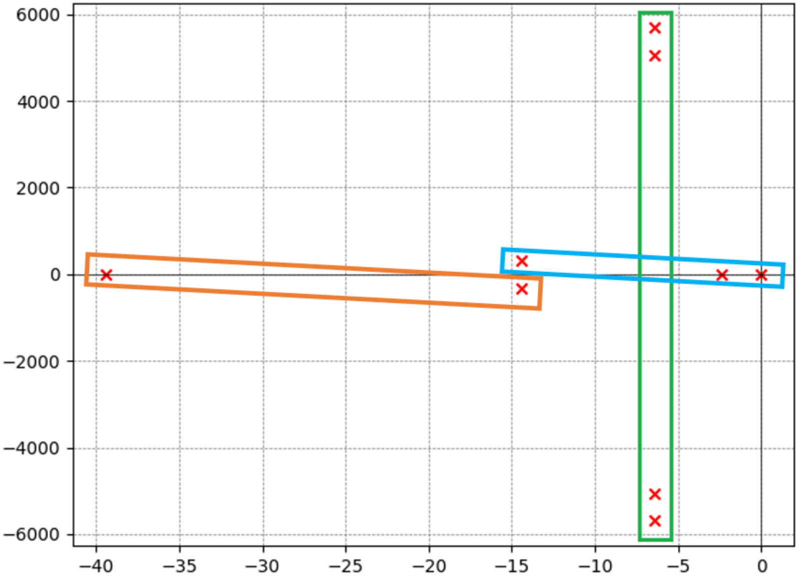
where Q is the matrix of eigenvectors and Λ is the diagonal matrix of eigenvalues. The spectral decomposition is then formulated as:

$$\Lambda = \Lambda_{s_1} + \Lambda_{s_2} + \Lambda_{s_3},$$

$$\begin{aligned}
\Lambda_{s_1} &= diag(\lambda_5, \lambda_8, \lambda_9, 0, 0, ..., 0), \\
\Lambda_{s_2} &= diag(0, 0, 0, \lambda_6, \lambda_7, 0, 0, ..., 0), \\
\Lambda_{s_3} &= diag(0, 0, ..., 0, \lambda_1, \lambda_2, \lambda_3, \lambda_4)
\end{aligned}$$

In each subspace s_k, we combine eigenvalues with low and high absolute value.

Plot of the eigenvalues

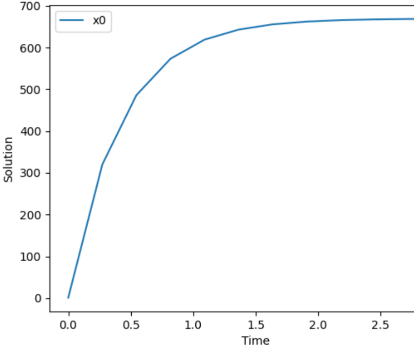


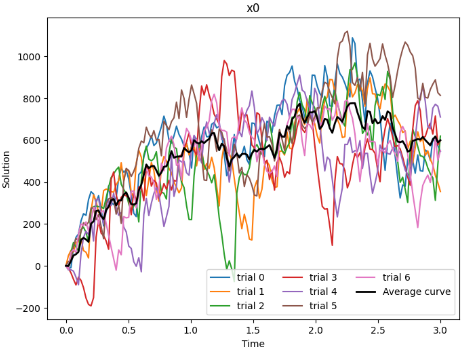
s_1: Subspace generated by the eigenvectors associated to $\{\lambda_6, \lambda_7\}$.
s_2: Subspace generated by the eigenvectors associated to $\{\lambda_5, \lambda_8, \lambda_9\}$.
s_3: Subspace generated by the eigenvectors associated to $\{\lambda_1, \lambda_2, \lambda_3, \lambda_4\}$.

$$
\begin{aligned}
A &= A_{s_1} + A_{s_2} + A_{s_3}, \\
A_{s_1} &= Q\Lambda_{s_1} Q^{-1}, \\
A_{s_2} &= Q\Lambda_{s_2} Q^{-1}, \\
A_{s_3} &= Q\Lambda_{s_3} Q^{-1}.
\end{aligned}
$$

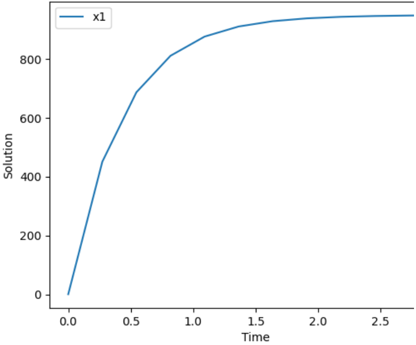
Solutions of the states using Radau IIA ($N = 150$)

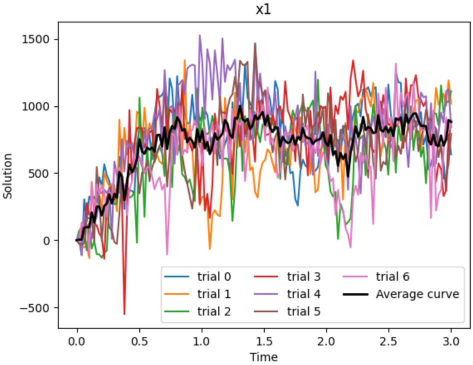
✓With Radau we obtain very similar solutions to the ode15s method in Matlab
Initial condition: steady-state point



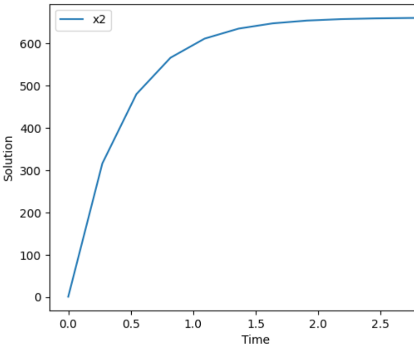


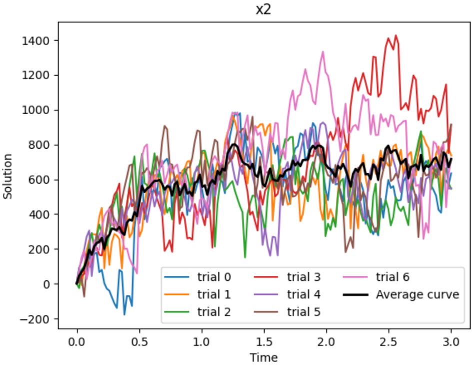
Solutions of the states using Radau IIA ($N = 150$)



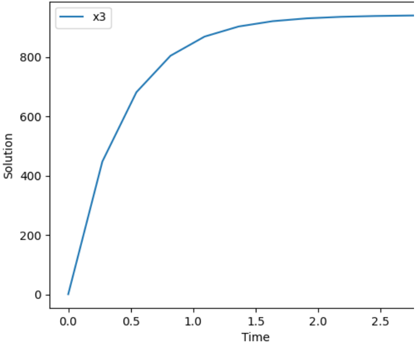


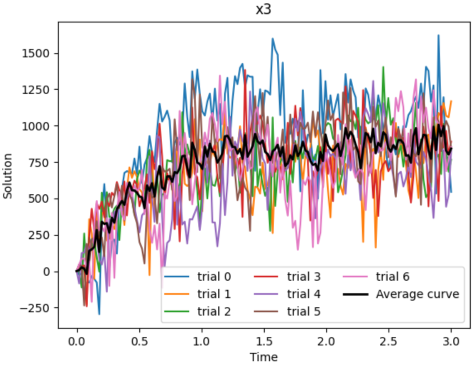
Solutions of the states using Radau IIA ($N = 150$)



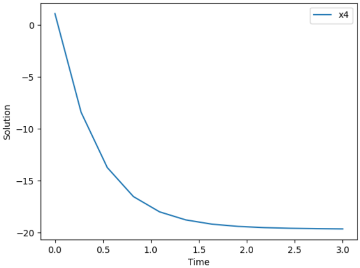


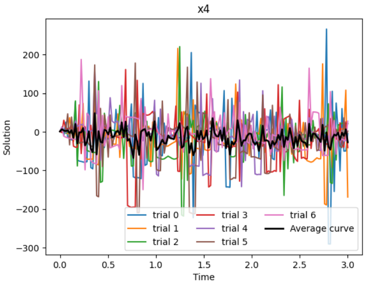
Solutions of the states using Radau IIA ($N = 150$)





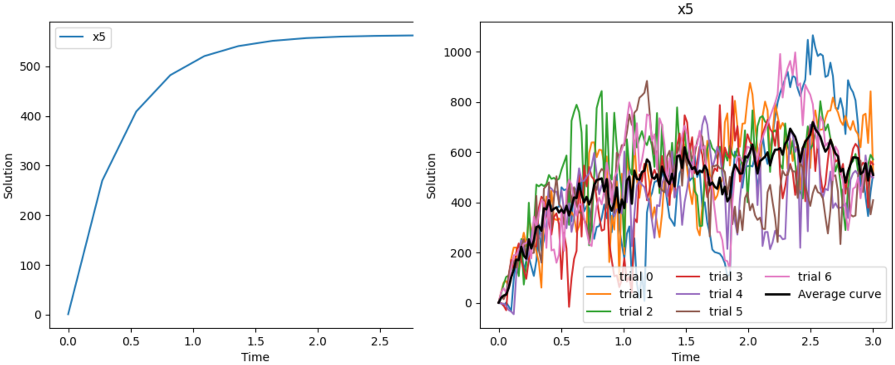
Solutions of the states using Radau IIA ($N = 150$)



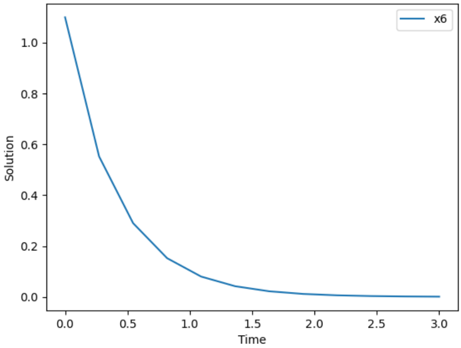


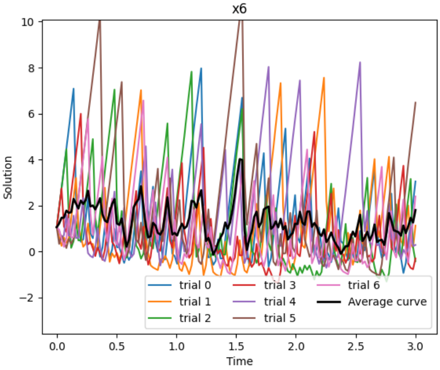
Not resembling the original solution curve

Solutions of the states using Radau IIA ($N = 150$)



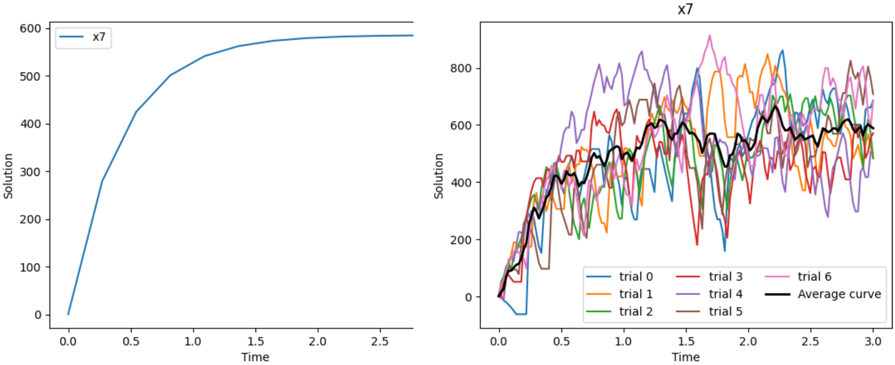
Solutions of the states using Radau IIA ($N = 150$)



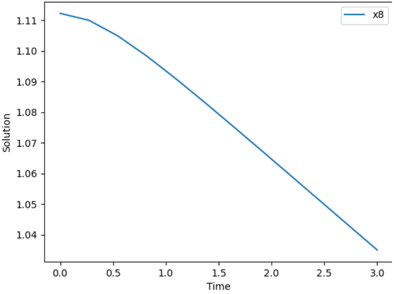


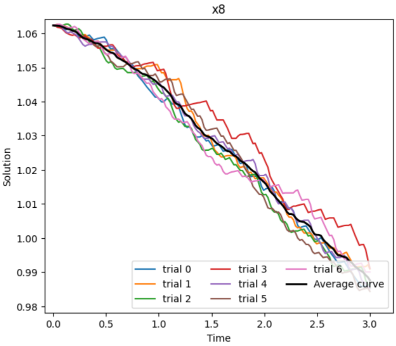
Not resembling the original solution curve

Solutions of the states using Radau IIA ($N = 150$)

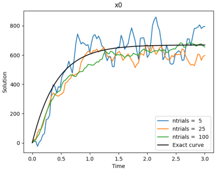


Solutions of the states using Radau IIA ($N = 150$)

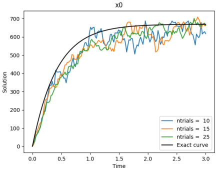




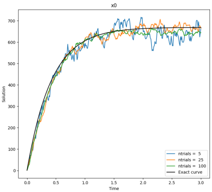
Convergence of the RBM with respect to ntrials and K



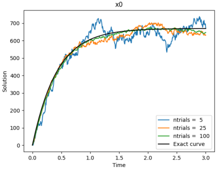
(a) $K = 100$



(b) $K = 150$



(c) $K = 300$



(d) $K = 500$

MPC Procedure

- We denote $\tau_i := \tau i$, $i \geq 0$, in the following.
- Starting at $i = 0$, we predict over $[0, T]$, and obtain an optimal control $\boldsymbol{u}_T^*(t)$, which minimizes

$$J_{T,\tau_0}(\boldsymbol{u}_T) = \int_0^T \left(\boldsymbol{x}_T(t)^T Q \boldsymbol{x}_T(t) + \boldsymbol{u}_T(t)^T R \boldsymbol{u}_T(t) \right) dt,$$

where $\boldsymbol{x}_T(t)$ fulfills

$$\dot{\boldsymbol{x}}_T(t) = A\boldsymbol{x}_T(t) + B\boldsymbol{u}_T(t), \quad \boldsymbol{x}_T(0) = \boldsymbol{x}_0. \tag{8}$$

- We now apply $\boldsymbol{u}_T^*$ to the true dynamics and obtain, like this, the state $\boldsymbol{x}_T^*$

$$\dot{\boldsymbol{x}}_T^*(t) = A\boldsymbol{x}_T^*(t) + B\boldsymbol{u}_T^*(t), \quad \boldsymbol{x}_T^*(0) = \boldsymbol{x}_0, \tag{9}$$

which we set to the MPC trajectory $\boldsymbol{x}_M^*$ on $t \in [0, \tau_1]$.
- This procedure is repeated: Starting from the state $\boldsymbol{x}_M^*(\tau_1)$, we predict the randomized control $\boldsymbol{u}_T^*$ over $[\tau_1, \tau_1 + T]$ and apply it to the system over $[\tau_1, \tau_1 + T]$, which yields $\boldsymbol{x}_T^*$ on $[\tau_1, \tau_2]$.

MPC Procedure Summary

1. Initialize the state: $\boldsymbol{x}_M^*(0) = \boldsymbol{x}_0$, $i = 0$.
2. **While** $\tau_i + T \leq T_{\max}$:
   a) Compute $\boldsymbol{u}_T^*(t, \boldsymbol{x}_M^{\tau_i})$ on $[\tau_i, \tau_i + T]$.
   b) Determine $\boldsymbol{x}_T^*(t, \boldsymbol{x}_M^{\tau_i})$ on $[\tau_i, \tau_{i+1}]$ by solving

$$\dot{\boldsymbol{x}}_T^*(t, \boldsymbol{x}_M^*(\tau_i)) = A\boldsymbol{x}_T^*(t, \boldsymbol{x}_M^*(\tau_i)) + B\boldsymbol{u}_T^*(t, \boldsymbol{x}_M^*(\tau_i)).$$

   c) Set $\boldsymbol{x}_M^*(t) = \boldsymbol{x}_T^*(t, \boldsymbol{x}_M^*(\tau_i))$ on $[\tau_i, \tau_{i+1}]$.
   d) $i = i + 1$.

Example. 1D heat equation with controls

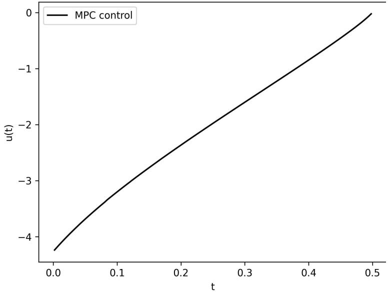
$$\frac{\partial y(t,\xi)}{\partial t} = \frac{\partial^2 y(t,\xi)}{\partial x^2} + \chi_{[-L/3,0]}(\xi)u(t), \ \xi \in [-L, L] \qquad (14)$$

$$\frac{\partial y(t,-L)}{\partial t} = \frac{\partial y(t,L)}{\partial t} = 0, \qquad (15)$$

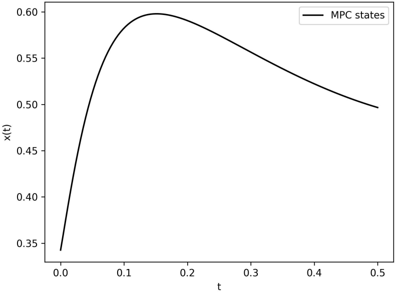
$$y(0,\xi) = \exp(-\xi^2) + \xi^2 \exp(-L^2). \qquad (16)$$

Plot for the MPC for the 1D heat equation with controls

$$T = 0.5, \tau = 0.05$$



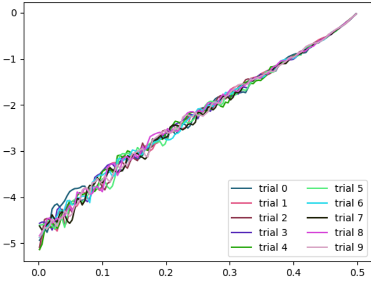
Controls



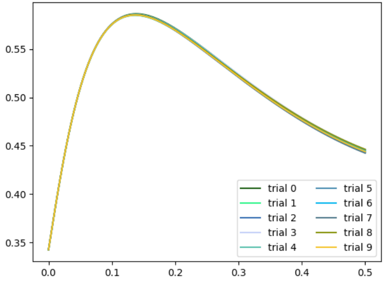
States

Plot for RBM + MPC for the 1D heat equation with controls

$T = 0.5, \tau = 0.05$, 9 realizations



Controls



States

Future work

- Simulation of the linealized system of the electrical model for more initial conditions.
- Physical interpretation of the constant controls used in the problem.
- Compare the state solutions of the linealized problem with the nonlinear one.
- Introduce non-constant controls, to apply the Model Predictive Control (MPC) to Converted-Dominated Power System.
- Follow the same methodology for more complex power systems, for example more controllers and more converters.

References

- Y. Gu, N. Bottrell & T. C. Green, Reduced-order models for representing converters in power system studies, IEEE Trans. Power Electron., 33.4 (2018), pp. 3644-54.
- U. Biccari, N. Sakamoto, E. Unamuno, D. Madariaga, E. Zuazua & J.A. Barrena, Model reduction of converter-dominated power systems by Singular Perturbation Theory, preprint (2019).
- Veldman, D. W., & Zuazua, E. (2022). A framework for randomized time-splitting in linear-quadratic optimal control. Numerische Mathematik, 151(2), 495-549.
- Ko, D., & Zuazua, E. (2021). Model predictive control with random batch methods for a guiding problem. Mathematical Models and Methods in Applied Sciences, 31(08), 1569-1592.