Koopman-BoxQP: Solving Large-Scale NMPC at kHz Rates

Liang Wu[1] WLIANG14@JH.EDU
Wallace Gian Yion Tan[2] WTGY@MIT.EDU
Richard D. Braatz[2] BRAATZ@MIT.EDU
Ján Drgoňa[1] JDRGONA1@JH.EDU
[1]*Johns Hopkins University, MD 21218, USA*
[2]*Massachusetts Institute of Technology, MA 02139, USA.*

Editors: G. Sukhatme, L. Lindemann, S. Tu, A. Wierman, N. Atanasov

Abstract

Solving large-scale nonlinear model predictive control (NMPC) problems at kilohertz (kHz) rates on standard processors remains a formidable challenge. This paper proposes a *Koopman-BoxQP* framework that *i)* learns a linear Koopman high-dimensional model, *ii)* eliminates the high-dimensional observables to construct a multi-step prediction model of the states and control inputs, *iii)* penalizes the multi-step prediction model into the objective, which results in a structured box-constrained quadratic program (BoxQP) whose decision variables include both the system states and control inputs, *iv)* develops a *structure-exploited* and *warm-starting-supported* variant of the *feasible* Mehrotra's interior-point algorithm for BoxQP. Numerical results demonstrate that *Koopman-BoxQP* can solve a large-scale NMPC problem with 1040 variables and 2080 inequalities at a kHz rate.

Keywords: Data-driven Koopman Operator, Nonlinear MPC, Quadratic Programming

1. Introduction

Modern quadratic programming (QP) solvers (Domahidi et al., 2012; Ferreau et al., 2014; Frison and Diehl, 2020; Zanelli et al., 2020; Wu and Bemporad, 2023) are able to solve small-scale linear MPC problems ($<$50 variables and $<$100 inequalities) at kHz rates (milliseconds) on standard processors, but fail to achieve similar performance for medium- or large-scale linear MPC problems (around $1000+$ variables and $2000+$ inequalities), let alone nonlinear programming-based nonlinear MPC problems (NMPC) (Diehl et al., 2009). Recently, data-driven Koopman approaches, which learn linear dynamics in a high-dimensional observable space, have emerged as a computationally efficient way to transform NMPC problems into compact general QP formulations by eliminating the high-dimensional observables (Korda and Mezić, 2018b; Folkestad et al., 2020).

While existing Koopman-MPC frameworks use the *condensed* MPC-to-QP construction (Korda and Mezić, 2018b) that only involves the control inputs (resulting in fewer decision variables), the number of state inequality constraints can become very large when the state dimension is high, as is commonly the case in control of partial differential equations (PDEs) (Arbabi et al., 2018). As the computation times of QP solvers typically depend on the *total number of variables and inequalities*, solving large-scale Koopman MPC problems at kHz rates using general-purpose QP solvers without exploiting problem structure remains computationally infeasible. Moreover, for practical use, soft-constrained MPC formulations (Zeilinger et al., 2014) that introduce slack variables for state constraints to guarantee feasibility (at the expense of higher computational cost due to the increased number of variables) are commonly employed to handle unknown disturbances and modeling errors. Modeling errors are inevitable, as data-driven finite-dimensional Koopman frameworks always involve projection and sampling errors (Zhang et al., 2022; Strässer et al., 2025).

This article proposes a dynamics-relaxed alternative for Koopman-MPC problems, which penalizes the approximated Koopman dynamic model via an ℓ_2-penalty term in the objective, rather than embedding it into the inequality constraints as in *condensed* Koopman's MPC-to-QP construction (Korda and Mezić, 2018b). The proposed dynamics-relaxed Koopman-MPC approach can be regarded as a *sparse* Koopman's MPC-to-QP construction, as the states are retained, while eliminating the high-dimensional observables in the offline phase. Unlike the *condensed* formulation in Korda and Mezić (2018b), which yields a general QP, the proposed dynamics-relaxed Koopman-MPC formulation results in a structured box-constrained QP (BoxQP). To tackle this problem efficiently, this article develops a *structure-exploited* and *warm-starting-supported* (note that the warm-starting does not work in IPMs for general QPs) Mehrotra's predictor–corrector interior point method (IPM) (Mehrotra, 1992), which is capable of solving the resulting large-scale BoxQP at kHz rates. The overall framework is referred to as the *Koopman-BoxQP* solution for NMPC problems.

1.1. Contributions

This article makes the following contributions, which are also the key factors as to how the proposed structure-exploited *Koopman-BoxQP* achieves kHz-rate performance:

1) *strictly feasible* cold- and warm-start initializations are provided cost-free, enabling direct use of the *feasible* Mehrotra's predictor–corrector IPM algorithm;
2) the *feasible* Mehrotra IPM typically converges to high-accuracy solutions of BoxQPs around $\mathbf{10}$ iterations (under strictly feasible cold-starts), regardless of the problem dimension, which is the key numerical finding highlighted in this article. Moreover, warm-starting performs well in the *feasible* Mehrotra IPM for solving BoxQPs and in real-time MPC problems; warm-starting from previous solutions halves the iterations to about $\mathbf{6}$;
3) by exploiting structure, the dimension of the linear system solved in each IPM step is reduced from $5(n_u+n_x)N_p$ to n_uN_p, yielding substantial speedups (when $n_x \gg n_u$, as in PDE control).

2. Problem Formulation

This article considers a nonlinear MPC problem (NMPC) for tracking, as shown in (1), where $x(t)$ is the feedback state at the sampling time t, $u_k \in \mathbb{R}^{n_u}$ and $x_k \in \mathbb{R}^{n_x}$ denote the control input and state at the kth time step, the prediction horizon length is N, x_r and u_r denote the desired tracking reference signal for the state and control input, respectively, and $W_x \succ 0$, $W_u \succ 0$, and $W_{\Delta u} \succ 0$ are weighting matrices for the deviation of the state tracking error, control input tracking error, and control input increment, respectively. Assume that W_x, W_u, and $W_{\Delta u}$ are diagonal. The equality constraint (1c) represents the nonlinear discrete-time dynamical system. Without loss of generality, the state and control input constraints, $[x_{\min}, x_{\max}]$ and $[u_{\min}, u_{\max}]$, are scaled to $[-\mathbf{1}, \mathbf{1}]$.

Nonlinear MPC:

$$\min \sum_{k=1}^{N-1} \|x_{k+1} - x_r\|^2_{W_x} + \|u_k - u_r\|^2_{W_u} + \|u_k - u_{k-1}\|^2_{W_{\Delta u}} \tag{1a}$$

$$\text{s.t. } x_0 = x(t),\ u_{-1} = 0, \tag{1b}$$

$$x_{k+1} = f(x_k, u_k), \quad k = 0, \dots, N-1 \tag{1c}$$

$$-\mathbf{1} \le x_{k+1} \le \mathbf{1}, \quad k = 0, \dots, N-1 \tag{1d}$$

$$-\mathbf{1} \le u_k \le \mathbf{1}, \quad k = 0, \dots, N-1 \tag{1e}$$

NMPC (1) is a nonconvex nonlinear program, which can be infeasible, can be sensitive to the noise-contaminated $x(t)$, can have slow solving speed, and can lack of convergence guarantees. A Koopman framework for MPC (Korda and Mezić, 2018b), which transforms NMPC (1) into a convex QP problem via data-driven Koopman approximations, allows the use of computationally efficient and convergent QP algorithms for real-time applications.

3. Preliminary: Koopman transforms NMPC into general QP

The Koopman operator (Koopman, 1931; Koopman and Neumann, 1932) provides a globally linear representation of nonlinear dynamics, enabling efficient linear techniques for nonlinear systems. Originally developed for autonomous systems, Koopman operator theory has since been extended to controlled systems of the form (1c) through various schemes (Williams et al., 2016; Proctor et al., 2018; Korda and Mezić, 2018b). In practice, the infinite-dimensional Koopman operator is truncated and approximated using data-driven Extended Dynamic Mode Decomposition (EDMD) methods (Williams et al., 2015, 2016; Korda and Mezić, 2018a,b). In EDMD specifically, the set of extended observables is designed as the "lifted" mapping, $\begin{bmatrix}\psi(x)\\ \mathbf{u}(0)\end{bmatrix}$, where $\mathbf{u}(0)$ denotes the first component of the sequence $\mathbf{u}$ and $\psi(x) \triangleq \left[\psi_1(x), \cdots, \psi_{n_\psi}(x)\right]^\top$ ($n_\psi \gg n_x$) is chosen from a basis function, e.g., Radial Basis Functions (RBFs) used in Korda and Mezić (2018b), instead of directly solving for them via optimization. In particular, the approximate Koopman operator identification problem is reduced to a least-squares problem, which assumes that the sampled data $\{(x_j, \mathbf{u}_j), (x_j^+, \mathbf{u}_j^+)\}$ (where j denotes the index of data samples and the superscript $+$ denotes the value at the next time step) are collected with the update mapping $\begin{bmatrix}x_j^+\\ \mathbf{u}_j^+\end{bmatrix} = \begin{bmatrix}f(x_j, \mathbf{u}_j(0))\\ \boldsymbol{S}\mathbf{u}_j\end{bmatrix}$. Then an approximation of the Koopman operator, $\mathcal{A} \triangleq [A \;\; B]$, can be obtained by solving

$$J(A,B) = \min_{A,B} \sum_{j=1}^{N_d} \|\psi(x_j^+) - A\psi(x_j) - B\mathbf{u}_j(0)\|_2^2. \tag{2}$$

According to Korda and Mezić (2018b), if the designed lifted mapping $\psi(x)$ contains the state x after the re-ordering $\psi(x) \leftarrow [x^\top, \psi(x)]^\top$, then $C = [I, 0]$. The learned linear Koopman predictor model is given as

$$\psi_{k+1} = A\psi_k + Bu_k, \; x_{k+1} = C\psi_{k+1}, \tag{3}$$

where $\psi_k \triangleq \psi(x_k) \in \mathbb{R}^{n_\psi}$ denotes the lifted state space and with $\psi_0 = \psi(x(t))$. If (3) has a high lifted dimension (for a good approximation), its use in MPC will not increase the dimension of the resulting QP if the high-dimensional observables ψ_k are eliminated via

$$\begin{bmatrix}x_1\\ x_2\\ \vdots\\ x_N\end{bmatrix} = \mathbf{E}\psi(x(t)) + \mathbf{F}\begin{bmatrix}u_0\\ u_1\\ \vdots\\ u_{N-1}\end{bmatrix}, \text{ where } \mathbf{E} \triangleq \begin{bmatrix}CA\\ CA^2\\ \vdots\\ CA^N\end{bmatrix}, \; \mathbf{F} \triangleq \begin{bmatrix}CB & 0 & \cdots & 0\\ CAB & CB & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ CA^{N-1}B & CA^{N-2}B & \cdots & CB\end{bmatrix}. \tag{4}$$

Then, by embedding (4) into the quadratic objective (1a) and the state constraint (1d), NMPC (1) can be reduced to a compact *general* QP with the decision vector $z \triangleq \mathrm{col}(u_0, \cdots, u_{N-1}) \in \mathbb{R}^{N\times n_u}$ as shown in (5), where $\bar{W}_x \triangleq \mathrm{blkdiag}(W_x, \cdots, W_x)$, $\bar{W}_u \triangleq \mathrm{blkdiag}(W_u, \cdots, W_u)$, $\bar{R} \triangleq$

$\text{blkdiag}(\bar{W}_{\Delta u}, \cdots, \bar{W}_{\Delta u}, \bar{W}^N_{\Delta u})$ $\left(\bar{W}_{\Delta u} = \begin{bmatrix} 2W_{\Delta u} & -W_{\Delta u} \\ -W_{\Delta u} & 2W_{\Delta u} \end{bmatrix} \text{ and } \bar{W}^N_{\Delta u} = \begin{bmatrix} 2W_{\Delta u} & -W_{\Delta u} \\ -W_{\Delta u} & W_{\Delta u} \end{bmatrix} \right)$ and $h \triangleq \mathbf{F}^\top \bar{W}_x(\mathbf{E}\psi(x(t)) - \bar{x}_r) - \bar{W}_u \bar{u}_r$ ($\bar{x}_r = \text{repmat}(x_r)$ and $\bar{u}_r = \text{repmat}(u_r)$). Note that the computation of the high-dimensional observable $\psi(x(t))$ is performed once and will not be involved in the iterations of QP, which minimizes a side effect of the high-dimensional Koopman operator.

NMPC → Koopman-QP:

$$\min_z \; z^\top \Big(\mathbf{F}^\top \bar{W}_x \mathbf{F} + \bar{W}_u + \bar{R} \Big) z + 2 z^\top h \tag{5a}$$
$$\text{s.t.} \; -\mathbf{1} \le \mathbf{E}\psi(x_t) + \mathbf{F} z \le \mathbf{1} \tag{5b}$$
$$-\mathbf{1} \le z \le \mathbf{1} \tag{5c}$$

The resulting Koopman-QP (5) has Nn_u decision variables and $2N(n_x + n_u)$ inequality constraints, with no dependence on n_ψ. The Koopman-QP (5) may be infeasible due to *inevitable* modeling errors in the data-driven Koopman *approximation*. By softening the state constraints (5b) to $-\mathbf{1} - \epsilon V_{\min} \le \mathbf{E}\psi(x_t) + \mathbf{F} z \le \mathbf{1} + \epsilon V_{\max}$ (the vectors $V_{\min}$ and $V_{\max}$ are non-positive) and adding a penalty term $\rho_\epsilon \epsilon^2$ ($\rho_\epsilon \gg W_x, W_u, W_{\Delta u}$) to the objective (5a), then the Koopman-QP (5) with the new decision vector $z \leftarrow \text{col}(z, \epsilon)$ is guaranteed to be feasible. Moreover, the Koopman-QP (5) constructed via condensing is a general QP without a specific structure, yet no existing work has developed a tailored structure-exploited QP solver for the Koopman-MPC framework to enable faster real-time computation.

4. Methodology

To simultaneously handle potential infeasibility and obtain faster execution time, this article proposes a *dynamics-relaxed* construction that transforms the Koopman-MPC problem into a parametric BoxQP, which is not always feasible but is guaranteed to be Lipschitz. The most important reason for choosing the construction is that we can customize a *structure-exploited* and *warm-starting-supported* IPM-based QP solver to solve the QP at kHz rates, which general QPs cannot do, as demonstrated in Subsection 4.2.

4.1. Koopman-BoxQP: Dynamics-relaxed Construction

This article proposes an alternative, the *dynamic relaxation approach*, which does not strictly enforce the Koopman prediction model (4) but instead incorporates it into the objective through a penalty term, which results in a strongly convex BoxQP (6), where $U \triangleq \text{col}(u_0, \cdots, u_{N-1})$, $X \triangleq \text{col}(x_1, \cdots, x_N)$, and $\rho > 0$ is a large penalty parameter reflecting the confidence in the Koopman model's accuracy.

Nonlinear MPC → *Koopman-BoxQP*

$$\begin{aligned} \min_{U,X} \; & (X - \bar{x}_r)^\top \bar{W}_x (X - \bar{x}_r) + (U - \bar{u}_r)^\top \bar{W}_u (U - \bar{u}_r) \\ & + U^\top \bar{R} U + \rho \| X - \mathbf{E}\psi\big(x(t)\big) - \mathbf{F} U \|_2^2 \\ \text{s.t.} \; & -\mathbf{1} \le U \le \mathbf{1}, \\ & -\mathbf{1} \le X \le \mathbf{1} \end{aligned} \tag{6}$$

For simplicity, we denote the decision vector $z \triangleq \text{col}(U, X) \in \mathbb{R}^n$ ($n = N(n_u + n_x)$), and the proposed

$$\begin{aligned} \min_z \; & \frac{1}{2} z^\top H z + z^\top h(x(t)) \\ \text{s.t.} \; & -\mathbf{1} \le z \le \mathbf{1} \end{aligned} \tag{7}$$

Koopman-BoxQP (6) is constructed as shown in (7), where

$$H \triangleq \rho \begin{bmatrix} \mathbf{F}^\top \mathbf{F} & -\mathbf{F}^\top \\ -\mathbf{F} & I \end{bmatrix} + \begin{bmatrix} \bar{W}_u + \bar{R} & \\ & \bar{W}_x \end{bmatrix} \succ 0, \quad h(x(t)) \triangleq \rho \begin{bmatrix} \mathbf{F}^\top \mathbf{E} \\ -\mathbf{E} \end{bmatrix} \psi(x(t)) - \begin{bmatrix} \bar{W}_u \bar{u}_r \\ \bar{W}_x \bar{x}_r \end{bmatrix}. \tag{8}$$

The dynamics-relaxed *Koopman-BoxQP* (6) can be viewed as an alternative approach to softening the state constraints, since the error in satisfying the prediction model can be equivalently interpreted as a relaxation of the state constraints.

The dynamics-relaxed *Koopman-BoxQP* (6) is a parametric optimization, which is proved below to be feasible and have a Lipschitz-continuous feedback policy. The former property prevents solver failure, and the latter property avoids chattering or abrupt control actions. Future work will leverage the proven Lipschitz constant of the parametric BoxQP, incorporating that of the Koopman observable, to certify the robustness of Koopman-MPC solutions (Teichrib and Darup, 2023).

Remark 1 (*Feasibility-guaranteed*) *The dynamics-relaxed Koopman-BoxQP* (6) *(or* (7)*) always admits a unique solution due to convexity and non-empty constraints, thus guaranteeing feasibility.*

Lemma 2 (*Lipschitz-guaranteed*) *Assume that the lifted mapping $\psi(\cdot)$ is Lipschitz continuous with constant L_ψ. The feedback policy of Koopman-BoxQP* (6)*,*

$$u_0(x(t)) \triangleq C_{policy} z^*(x(t)), \tag{9}$$

where $C_{policy} = [I_{n_u}, 0]$), is Lipschitz continuous with constant $\frac{\rho\sqrt{\lambda_{\max}(\mathbf{E}^\top(\mathbf{F}\mathbf{F}^\top+I)\mathbf{E})}L_\psi}{\lambda_{\min}(H)}$.

Proof The optimal solution $z^*(x(t))$ can be characterized by the variational inequality:

$$\big(Hz^*(x(t)) + h(x(t))\big)^\top \big(z - z^*(x(t))\big), \ \forall z \in [-\mathbf{1}, \mathbf{1}]. \tag{10}$$

Let us take $x(t_1), x(t_2)$ with corresponding solutions $z_1 \triangleq z^*(x(t_1))$, $z_2 \triangleq z^*(x(t_2))$ and define $\Delta h \triangleq h(x(t_2)) - h(x(t_1))$, $\Delta z \triangleq z_2 - z_1$, $\Delta x \triangleq x(t_2) - x(t_1)$. Applying (10) at each solution—for z_1 : $(Hz_1 + h(x(t_1)))^\top (z_2 - z_1) \geq 0$, for z_2 : $(Hz_2 + h(x(t_2)))^\top (z_1 - z_2) \geq 0$—and then adding the two inequalities results in

$$(Hz_1 + h(x(t_1)))^\top (z_2 - z_1) + (Hz_2 + h(x(t_2)))^\top (z_1 - z_2) \geq 0,$$

that is, $\Delta z^\top H \Delta z \leq -\Delta z^\top \Delta h$. $H \succ 0$ implies the strong convexity inequality $\lambda_{\min}(H)\|\Delta z\|_2^2 \leq \Delta z^\top H \Delta z$. By the Cauchy-Schwarz inequality and the assumption, $|\Delta z^\top \Delta h| \leq \|\Delta z\|_2 \|\Delta h\|_2$ and

$$\|\Delta h\|_2 \leq \rho \|[\mathbf{F}^\top \mathbf{E}; -\mathbf{E}]\|_2 L_\psi \|\Delta x\|_2 = \rho\sqrt{\lambda_{\max}(\mathbf{E}^\top(\mathbf{F}\mathbf{F}^\top + I)\mathbf{E})} L_\psi \|\Delta x\|_2.$$

Combining those inequalities gives that

$$\lambda_{\min}(H)\|\Delta z\|_2^2 \leq \|\Delta z\|_2 \rho\sqrt{\lambda_{\max}(\mathbf{E}^\top(\mathbf{F}\mathbf{F}^\top + I)\mathbf{E})} L_\psi \|\Delta x\|_2.$$

That is, $\|\Delta z\|_2 \leq \frac{\rho\sqrt{\lambda_{\max}(\mathbf{E}^\top(\mathbf{F}\mathbf{F}^\top+I)\mathbf{E})}L_\psi}{\lambda_{\min}(H)} \|\Delta x\|_2$, namely,

$$\|u_0(x(t_2)) - u_0(x(t_1))\|_2 \leq \|C_{\text{policy}}\| \|\Delta z\|_2 \leq \frac{\rho\sqrt{\lambda_{\max}(\mathbf{E}^\top(\mathbf{F}\mathbf{F}^\top + I)\mathbf{E})} L_\psi}{\lambda_{\min}(H)} \|x(t_2) - x(t_1)\|_2,$$

which completes the proof. ■

4.2. Feasible Mehrotra's IPM Algorithm for BoxQP

KKT condition of BoxQP

$$Hz + h(x(t)) + \gamma - \theta = 0, \quad \text{(11a)}$$
$$z + \phi - \mathbf{1}_n = 0, \quad \text{(11b)}$$
$$z - \psi + \mathbf{1}_n = 0, \quad \text{(11c)}$$
$$(\gamma, \theta, \phi, \psi) \geq 0, \quad \text{(11d)}$$
$$\gamma \odot \phi = 0, \quad \text{(11e)}$$
$$\theta \odot \psi = 0, \quad \text{(11f)}$$

According to Boyd and Vandenberghe (2004, Ch 5), the Karush–Kuhn–Tucker (KKT) condition of the BoxQP (7) is (11), where γ, θ are the Lagrangian variables of the lower and upper bound, respectively; ϕ, ψ are the slack variables of the lower and upper bound, respectively; $\odot$ is the Hadamard product, i.e., $\gamma \odot \phi = \mathrm{col}(\gamma_1\phi_1, \gamma_2\phi_2, \cdots, \gamma_n\phi_n)$. The BoxQP (7) may be ill-conditioned since the penalty parameter ρ needs to be large to ensure strong closed-loop performance. To address this issue, this article employs second-order IPMs, with a particular focus on the most computationally efficient variant: Mehrotra's predictor–corrector IPM (Mehrotra, 1992). In practice, Mehrotra's predictor–corrector IPM typically achieves highly accurate solutions within $50 \sim 75$ iterations, regardless of whether the initial point is strictly feasible. This remarkably fast convergence makes the method the foundation of most IPM software packages.

An important numerical observation in this article is that Mehrotra's predictor–corrector IPM typically takes *about 10 iterations* to find a highly accurate solution of BoxQP (7) when applying the following cost-free initialization strategy. To demonstrate this, denote the feasible region by $\mathcal{F}$, i.e., $\mathcal{F} = \{(z, \gamma, \theta, \phi, \psi) : \text{(11a)}-\text{(11c)}, (\gamma, \theta, \phi, \psi) \geq 0\}$ and the set of strictly feasible points by $\mathcal{F}^+ \triangleq \{(z, \gamma, \theta, \phi, \psi) : \text{(11a)}-\text{(11c)}, (\gamma, \theta, \phi, \psi) > 0\}$. For a point $\in \mathcal{F}^+$, the residuals of (11a)–(11c) are zeros and only the residuals of the complementary equations (11e)–(11f) are not.

4.3. Cost-free strictly feasible cold- and warm-starting initializations

IPMs for general optimization problems, including QPs, generally do not support warm-start initialization, which is an unfortunate limitation as warm-start initialization can substantially accelerate convergence and reduce iteration counts in real-time MPC applications. A notable advantage of the proposed *Koopman-BoxQP* (6) is its ability to support strictly feasible cold- and warm-start initializations without any extra computational overhead.

Remark 3 ***(Strictly feasible cold-start)**: Inspired from (Wu and Braatz, 2025, 2026), the initial point*

$$(z^0, \gamma^0, \theta^0, \phi^0, \psi^0) = (0, \|h(x(t))\|_\infty \mathbf{1}_n - \tfrac{1}{2}h(x(t)), \|h(x(t))\|_\infty \mathbf{1}_n + \tfrac{1}{2}h(x(t)), \mathbf{1}_n, \mathbf{1}_n) \in \mathcal{F}^+. \quad (12)$$

Remark 4 ***(Strictly feasible warm-start)**: The optimal solution at the previous sampling time $t-1$: $U^{t-1} = \mathrm{col}(u_0^{t-1}, \cdots, u_{N-1}^{t-1})$, $X^{t-1} = \mathrm{col}(x_1^{t-1}, \cdots, x_N^{t-1})$, can be shifted ahead one step to obtain a warm start initial point: $z^{t,guess} \triangleq \mathrm{col}(u_1^{t-1}, \cdots, u_{N-1}^{t-1}, u_{N-1}^{t-1}, x_2^{t-1}, \cdots, x_N^{t-1}, x_N^{t-1})$ (Clearly, $-\mathbf{1}_n \leq z^{t,guess} \leq \mathbf{1}_n$) to be used in solving the Koopman-BoxQP (6) at the current sampling time t. The warm start initial point is*

$$z^0 = z^{t,guess},\ \gamma^0 = \|\bar{h}\|_\infty \mathbf{1}_n - \tfrac{1}{2}\bar{h},\ \theta^0 = \|\bar{h}\|_\infty \mathbf{1}_n + \tfrac{1}{2}\bar{h},\ \phi^0 = \mathbf{1}_n - z^{t,guess},\ \psi^0 = \mathbf{1}_n + z^{t,guess}, \quad (13)$$

where $\bar{h} \triangleq Hz^{t,guess} + h(x(t))$. Notably, we have $(z^0, \gamma^0, \theta^0, \phi^0, \psi^0) \in \mathcal{F}^+$.

4.4. Algorithm description and efficient computation of Newton systems

Algorithm 1 presents a *feasible* variant of Mehrotra's predictor–corrector IPM, specifically tailored for BoxQP (7). At each iteration, Steps 3 and 4 of Algorithm 1 compute the predictor and corrector directions, respectively, using the same coefficient matrix J_k; thus, only a single matrix factorization is required. Moreover, the structured sparsity of J_k enables a more computationally efficient Cholesky factorization on a reduced-dimensional linear system, from $5N \times (n_u + n_x) \to N \times n_u$. Considering (14) and (16), which differ only in the last two terms on the right-hand side, denote these terms as r^1 and r^2, respectively. Then, by letting

$$\Delta\gamma = \frac{\gamma^k}{\phi^k}\Delta z + \frac{r^1}{\phi^k},\ \Delta\theta = -\frac{\theta^k}{\psi^k}\Delta z + \frac{r^2}{\psi^k},$$
$$\Delta\phi = -\Delta z,\ \Delta\psi = \Delta z, \tag{17}$$

Equation (14) (or (16)) can be reduced to a more compact system of linear equations,

$$\bar{H}\Delta z = \frac{r^1}{\phi} - \frac{r^2}{\psi}, \tag{18}$$

where the coefficient matrix $\bar{H}$ has a 2×2 block structure from the definition of H in (8):

$$\bar{H} \triangleq H + D\left(\frac{\gamma^k}{\phi^k} + \frac{\theta^k}{\psi^k}\right) = \begin{bmatrix} \bar{H}_{11} & -\rho\mathbf{F}^\top \\ -\rho\mathbf{F} & \bar{H}_{22} \end{bmatrix} \succ 0 \tag{19}$$

(where $D(\cdot)$ denotes the diagonal matrix of a vector) with

$$\bar{H}_{11} \triangleq \rho\mathbf{F}^\top\mathbf{F} + \bar{W}_u + \bar{R} + D\left(\frac{\gamma^k_{1:n_1}}{\phi^k_{1:n_1}} + \frac{\theta^k_{1:n_1}}{\psi^k_{1:n_1}}\right)$$
$$\bar{H}_{22} \triangleq \rho I + \bar{W}_x + D\left(\frac{\gamma^k_{n_1+1:n}}{\phi^k_{n_1+1:n}} + \frac{\theta^k_{n_1+1:n}}{\psi^k_{n_1+1:n}}\right)$$

where $\bar{H}_{11} \succ 0 \in \mathbb{R}^{n_1\times n_1}$, $\bar{H}_{22} \succ 0 \in \mathbb{R}^{n_2\times n_2}$, $n_1 = Nn_u$, and $n_2 = Nn_x$. By assumption the weighting matrix W_x is diagonal, thus $\bar{W}_x$ and $\bar{H}_{22}$ are also diagonal. By exploiting the diagonal structure of $\bar{H}_{22}$, computational savings in solving (18) can be achieved through solving the system:

$$\left(\bar{H}_{11} - \rho^2\mathbf{F}^\top\bar{H}_{22}^{-1}\mathbf{F}\right)\Delta z_{1:n_1} = \frac{r^1_{1:n_1}}{\phi^k_{1:n_1}} - \frac{r^2_{1:n_1}}{\psi^k_{1:n_1}} + \rho\mathbf{F}^\top\bar{H}_{22}^{-1}\left(\frac{r^1_{n_1+1:n}}{\phi^k_{n_1+1:n}} - \frac{r^2_{n_1+1:n}}{\psi^k_{n_1+1:n}}\right) \tag{20}$$

Algorithm 1 Feasible Mehrotra's predictor-corrector IPM for BoxQP (7)

Input: Initializing $(z^0, \gamma^0, \theta^0, \phi^0, \psi^0)$ from Eqn. (12), a desired optimal level ϵ, and the maximum number of iterations $N_{\max}$.

for $k = 0, 1, 2, \cdots, N_{\max} - 1$ **do**

1. $\mu^k \leftarrow [(\gamma^k)^\top\phi^k + (\theta^k)^\top\psi^k]/(2n)$;
2. if $\mu^k \le (2n)\epsilon$, then break;
3. Compute $(\Delta z^{\text{aff}}, \Delta\gamma^{\text{aff}}, \Delta\theta^{\text{aff}}, \Delta\phi^{\text{aff}}, \Delta\psi^{\text{aff}})$ by solving
$$J^k \begin{bmatrix} \Delta z^{\text{aff}} \\ \Delta\gamma^{\text{aff}} \\ \Delta\theta^{\text{aff}} \\ \Delta\phi^{\text{aff}} \\ \Delta\psi^{\text{aff}} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -\gamma^k\odot\phi^k \\ -\theta^k\odot\psi^k \end{bmatrix} \tag{14}$$
where
$$J^k \triangleq \begin{bmatrix} H & I & -I & 0 & 0 \\ I & 0 & 0 & I & 0 \\ I & 0 & 0 & 0 & -I \\ 0 & D(\phi^k) & 0 & D(\gamma^k) & 0 \\ 0 & 0 & D(\psi^k) & 0 & D(\theta^k) \end{bmatrix} \tag{15}$$
4. $\alpha^{\text{aff}} = \min(1, 0.99\min_{\Delta\gamma_i^{\text{aff}}<0}\frac{-\gamma_i^k}{\Delta\gamma_i^{\text{aff}}}, 0.99\min_{\Delta\theta_i^{\text{aff}}<0}\frac{-\theta_i^k}{\Delta\theta_i^{\text{aff}}}, 0.99\min_{\Delta\phi_i^{\text{aff}}<0}\frac{-\phi_i^k}{\Delta\phi_i^{\text{aff}}}, 0.99\min_{\Delta\psi_i^{\text{aff}}<0}\frac{-\psi_i^k}{\Delta\psi_i^{\text{aff}}})$;
5. $\mu^{\text{aff}} = [(\gamma^k + \alpha^{\text{aff}}\Delta\gamma^{\text{aff}})^\top(\phi^k + \alpha^{\text{aff}}\Delta\phi^{\text{aff}}) + (\theta^k + \alpha^{\text{aff}}\Delta\theta^{\text{aff}})^\top(\psi^k + \alpha^{\text{aff}}\Delta\psi^{\text{aff}})]/(2n)$;
6. $\sigma \leftarrow \left(\frac{\mu^{\text{aff}}}{\mu^k}\right)^3$;
7. Compute $(\Delta z, \Delta\gamma, \Delta\theta, \Delta\phi, \Delta\psi)$ by solving
$$J^k \begin{bmatrix} \Delta z \\ \Delta\gamma \\ \Delta\theta \\ \Delta\phi \\ \Delta\psi \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -\gamma^k\odot\phi^k - \Delta\gamma^{\text{aff}}\odot\Delta\phi^{\text{aff}} + \sigma\mu^k\mathbf{1}_n \\ -\theta^k\odot\psi^k - \Delta\theta^{\text{aff}}\odot\Delta\psi^{\text{aff}} + \sigma\mu^k\mathbf{1}_n \end{bmatrix} \tag{16}$$
8. $\alpha = \min(1, 0.99\min_{\Delta\gamma_i<0}\frac{-\gamma_i^k}{\Delta\gamma_i}, 0.99\min_{\Delta\theta_i<0}\frac{-\theta_i^k}{\Delta\theta_i}, 0.99\min_{\Delta\phi_i<0}\frac{-\phi_i^k}{\Delta\phi_i}, 0.99\min_{\Delta\psi_i<0}\frac{-\psi_i^k}{\Delta\psi_i})$;
9. $z^{k+1} \leftarrow z^k + \alpha\Delta z, \leftarrow \gamma^k + \alpha\Delta\gamma, \leftarrow \theta^k + \alpha\Delta\theta, \leftarrow \phi^k + \alpha\Delta\phi,\ \psi^{k+1} \leftarrow \psi^k + \alpha\Delta\psi$;

end

Output: z^{k+1}.

and $\Delta z_{n_1+1:n} = \bar{H}_{22}^{-1}\left(\frac{r^1_{n_1+1:n}}{\phi^k_{n_1+1:n}} - \frac{r^2_{n_1+1:n}}{\psi^k_{n_1+1:n}} + \rho\mathbf{F}\Delta z_{1:n_1}\right)$ to obtain the solution $\Delta z = \mathrm{col}(\Delta z_{1:n_1}, \Delta z_{n_1+1:n})$.

Remark 5 *By the Schur Complement lemma, $\bar{H}_{11} - \rho^2\mathbf{F}^\top\bar{H}_{22}^{-1}\mathbf{F} \succ 0$, and the matrix is symmetric. Then, the Cholesky factorization with the cost $O((Nn_u)^3)$ for (20) can be applied, which is crucial for achieving computational speedup.*

5. Numerical Examples

5.1. Practical behavior of cold-starting Mehrotra's IPM algorithm on random BoxQPs

The main inspiration for using the Koopman-BoxQP formulation comes from our contributed numerical observation as follows,

> *Feasible Mehrotra's IPM algorithm usually takes $\sim$ 10 iterations on (ill-conditioned) random BoxQP problems, regardless of the problem dimension.*

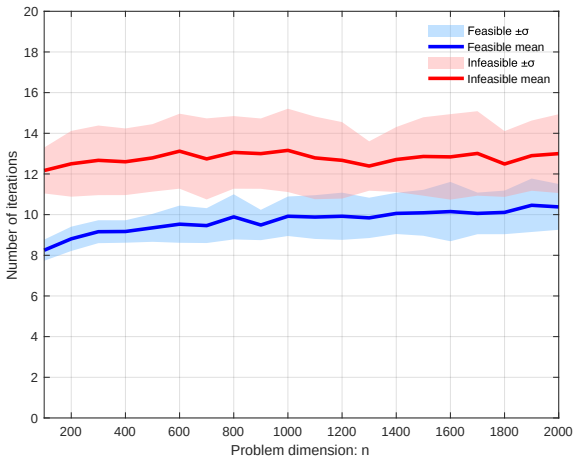

Figure 1: Iteration counts of feasible and infeasible Mehrotra's IPM algorithms for ill-conditioned random BoxQP problems with dimensions ranging from 100 to 2000.

To verify this observation, we generate random BoxQP problems with dimensions n ranging from 100 to 2000, where the Hessian matrix has a condition number of 10^6. Algorithm 1, with the desired optimality tolerance set $\epsilon = 10^{-6}$, is applied both with and without the cold-start initialization strategy, corresponding to the feasible and infeasible variants, respectively, for solving the generated BoxQP instances. For each specified problem dimension, the BoxQP is randomly generated and solved 100 times to characterize the statistical behavior of the iteration counts of feasible and infeasible Mehrotra's IPM algorithms, which are plotted in Fig. 1. Clearly, Fig. 1 confirms our key numerical observation: the feasible Mehrotra's IPM algorithm (Algorithm 1) is **scalable**, requiring only about 10 iterations to achieve highly accurate solutions.

5.2. Performance validation of *Koopman-BoxQP* on PDE NMPC problems

This section applies our proposed *Koopman-BoxQP* to solve a PDE-MPC problem with state and control input constraints, which is a large-sized QP problem with 1040 variables and 2080 constraints. The PDE plant under consideration is the nonlinear Korteweg-de Vries (KdV) equation that models the propagation of acoustic waves in plasma or shallow water waves (Miura, 1976) as

$$\frac{\partial y(t,x)}{\partial t} + y(t,x)\frac{\partial y(t,x)}{\partial x} + \frac{\partial^3 y(t,x)}{\partial x^3} = u(t,x) \tag{21}$$

where $x \in [-\pi, \pi]$ is the spatial variable. Consider the control input $u(t,x) = \sum_{i=1}^{4} u_i(t)v_i(x)$, in which the four coefficients $\{u_i(t)\}$ are subject to the constraint $[-1, 1]$, and $v_i(x)$ are predetermined

spatial profiles given as $v_i(x) = e^{-25(x-m_i)^2}$, with $m_1 = -\pi/2$, $m_2 = -\pi/6$, $m_3 = \pi/6$, and $m_4 = \pi/2$. The control objective is to adjust $u_i(t)$ so that the spatial profile $y(t,x)$ tracks the given reference signal. The spatial profile $y(t,x)$ is uniformly discretized into 100 spatial nodes. These 100 discretized nodes serve as the system states, each constrained within $[-1,1]$, while the control inputs are likewise bounded within $[-1,1]$. Choosing the MPC prediction horizon as $N = 10$ results in a medium-scale optimization problem with 1040 variables and 2080 constraints. For the MPC settings, the state cost matrix is set to $W_x = I_{100}$, and the control inputs matrix is set to $W_u = 0.05I_4$. The state references $x_r \in \mathbb{R}^{100}$ are sinusoidal signals for a 50 s simulation time, and the control input reference is a constant value with $u_r = 0$. We employ a spectral method based on the Fourier transform and a split-step scheme to solve the nonlinear KdV (21), for data generation and MPC closed-loop simulation. The sampling time is chosen to be $\Delta t = 0.01$.

The setting for our closed-loop simulation includes: *(i) Data generation:* The data are collected from 1000 simulation trajectories with 200 samples. At each simulation, the initial condition of the spatial profile is a random combination of four given spatial profiles, i.e., $y_1^0(0,x) = e^{-(x-\pi/2)^2}$, $y_2^0(0,x) = -\sin(x/2)^2$, $y_3^0(0,x) = e^{-(x+\pi/2)^2}$, $y_4^0(0,x) = \cos(x/2)^2$. The four control inputs $u_i(t)$ are distributed uniformly in $[-1,1]$; *(ii) Koopman predictor:* The lift function ψ is composed of the original 100 spatial states and 200 thin-plate RBFs with random centers, which leads to the lifted state dimension $N_\psi = 100 + 200 = 300$. Then the lifted linear predictor with $A \in \mathbb{R}^{300\times300}$ and $B \in \mathbb{R}^{300\times4}$ is obtained from the Moore-Penrose pseudoinverse of the lifting data matrix, and its output matrix is $C = [I_{100}, 0] \in \mathbb{R}^{100\times300}$; *(iii) Koopman-BoxQP formulation:* Given the matrices $\{A, B, C\}$ calculated from the above previous Koopman predictor step, the approximated multi-step Koopman model (4) is constructed and embedded into the *Koopman-BoxQP* (6) with dynamic penalty parameter $\rho = 10^2$. This results in a BoxQP with 1040 variables and 2080 constraints.

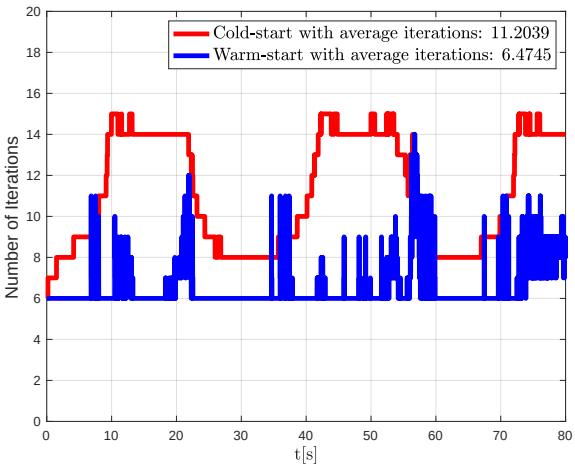

Figure 2: Iterations comparison of Algorithm 1 with cold-start and warm-start.

QP Solver	Average Execution Time [s][1] cold start	warm start
Quadprog	0.1127	0.1127
OSQP	9.7×10^{-3}	1.2×10^{-3}
SCS	5.8×10^{-3}	5.8×10^{-3}
Algorithm 1	$\mathbf{5.57\times10^{-4}}$	$\mathbf{3.63\times10^{-4}}$

Table 1: Execution time comparison between Algorithm 1 and other state-of-the-art QP solvers.

The proposed Algorithm 1 with the stopping criteria $\epsilon = 1\times10^{-6}$ is applied to solve the resulting large-size *Koopman-BoxQP* problem (6) in the closed-loop simulation. The closed-loop performance, shown in Fig. 3, demonstrates that the spatial profile $y(t,x)$ accurately tracks the desired reference signals while remaining within the state constraint $[-1,1]$, even when the reference signals exceed this bound. The control inputs

1. The execution time results are based on MATLAB's C-MEX implementation of Algorithm 1 running on a Mac mini with an Apple M4 Chip (10-core CPU and 16 GB RAM).

also remain strictly within their limits of $[-1, 1]$. This numerical case study demonstrates that the proposed *Koopman-BoxQP* approach can accurately control the evolution of large-scale states.

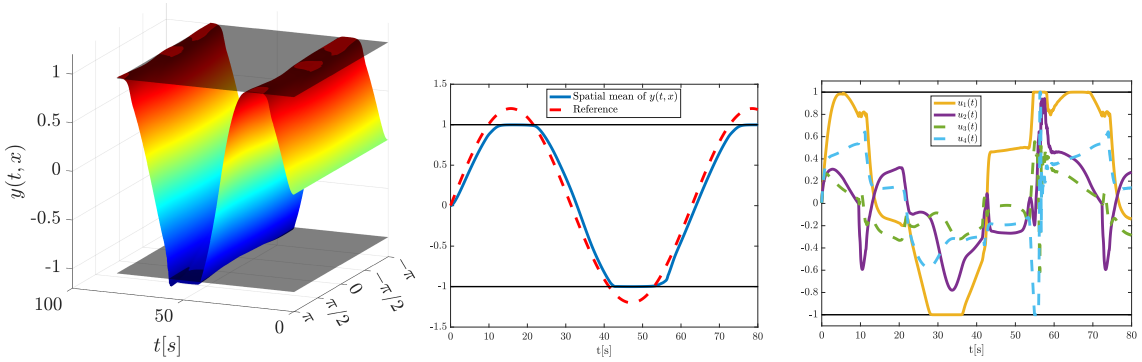

Figure 3: Closed-loop simulation of the nonlinear KdV system with the dynamics-relaxed *Koopman-BoxQP* controller tracking a time-varying spatial profile reference. Left: time evolution of the spatial profile $y(t, x)$ and the state constraints $[-1, 1]$. Middle: spatial mean of the $y(t, x)$ and the state constraints $[-1, 1]$. Right: the four control inputs and the control input constraints $[-1, 1]$.

The number of iterations of the IPM-based Algorithm 1 with cold and warm starts is plotted in Fig. 2, which demonstrates that Algorithm 1 effectively supports warm-start initialization, reducing the average number of iterations by half, from $\mathbf{11.2039}$ (cold start) to $\mathbf{6.4745}$ (warm start). This implies that solving around $\mathbf{6}$ small-scale linear systems (reduced from large-scale via exploiting the problem structure) can obtain 10^{-6}-accuracy solutions, *explaining why Algorithm 1 can solve large-scale NMPC problems at kHz rates (in milliseconds)* as shown in Table 1. Other state-of-the-art QP solvers, including MATLAB's Quadprog (using IPM), OSQP (Stellato et al., 2020), and SCS (O'Donoghue, 2021) (all choose the 'eps_rel', '1e-6' for a fair comparison), are compared for solving the same resulting *Koopman-BoxQP* (6). Their average execution times under cold- and warm-start initialization are listed in Table 1, which shows that using a warm start is not helpful in the IPM-based solvers Quadprog and SCS, but beneficial in Algorithm 1. Furthermore, Algorithm 1 is the only solver that achieves kHz-level solution speeds for large-scale NMPC problems. Moreover, the proposed Algorithm 1 has been extended to general QPs, and more numerical experiments can be found in the Appendix A due to space limits.

6. Conclusion

This article presents a fast *Koopman-BoxQP* framework enabling NMPC at kHz rates. Our approach consists of a *dynamics-relaxed* BoxQP formulation and a *linear-algebra-tailored* and *warm-starting-supported* IPM-based QP solver. We validated the approach on a challenging large-sized PDE-MPC problem (with 1040 decision variables and 2080 inequality constraints) and achieved a real-time solution under < 1 ms on a standard desktop CPU, opening a new era of kHz-rate solutions for large-scale NMPC. Future work will focus on developing a GPU-accelerated *Koopman-BoxQP* solver, leveraging the computational power of cuBLAS and cuSOLVER for large-scale linear system solves, together with the practical scalable iteration complexity (typically around 10 iterations) of the IPM in BoxQPs, to enable larger PDE-MPC applications. The stability proof and robustness of the *dynamics-relaxed Koopman-BoxQP* will also be investigated.

Acknowledgments

This research was supported by the Ralph O'Connor Sustainable Energy Institute at Johns Hopkins University, and by the U.S. DOE, Office of Science, ASCR program under the Scientific Discovery through Advanced Computing (SciDAC) Institute "LEADS: LEarning-Accelerated Domain Science". Wallace Tan was supported by the MathWorks Fellowship. Richard Braatz was supported by the U.S. Food and Drug Administration under the FDA BAA-22-00123 program (Award 75F40122C00200).

References

Hassan Arbabi, Milan Korda, and Igor Mezić. A Data-Driven Koopman Model Predictive Control Framework for Nonlinear Partial Differential Equations. In *IEEE Conference on Decision and Control*, pages 6409–6414, 2018.

Stephen P. Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, U.K., 2004.

Moritz Diehl, Hans Joachim Ferreau, and Niels Haverbeke. Efficient numerical methods for nonlinear MPC and moving horizon estimation. In *Nonlinear Model Predictive control: Towards New Challenging Applications*, pages 391–417. Springer Verlag, 2009.

Alexander Domahidi, Aldo U. Zgraggen, Melanie N. Zeilinger, Manfred Morari, and Colin N. Jones. Efficient interior point methods for multistage problems arising in receding horizon control. In *IEEE 51st IEEE Conference on Decision and Control*, pages 668–674, 2012.

Hans Joachim Ferreau, Christian Kirches, Andreas Potschka, Hans Georg Bock, and Moritz Diehl. qpOASES: A parametric active-set algorithm for quadratic programming. *Mathematical Programming Computation*, 6(4):327–363, 2014.

Carl Folkestad, Daniel Pastor, Igor Mezic, Ryan Mohr, Maria Fonoberova, and Joel Burdick. Extended dynamic mode decomposition with learned Koopman eigenfunctions for prediction and control. In *American Control Conference*, pages 3906–3913, 2020.

Gianluca Frison and Moritz Diehl. HPIPM: A high-performance quadratic programming framework for model predictive control. *IFAC-PapersOnLine*, 53(2):6563–6569, 2020.

Bernard O. Koopman. Hamiltonian systems and transformation in Hilbert space. *Proceedings of the National Academy of Sciences*, 17(5):315–318, 1931.

Bernard O. Koopman and J. V. Neumann. Dynamical systems of continuous spectra. *Proceedings of the National Academy of Sciences*, 18(3):255–263, 1932.

Milan Korda and Igor Mezić. On convergence of extended dynamic mode decomposition to the Koopman operator. *Journal of Nonlinear Science*, 28(2):687–710, 2018a.

Milan Korda and Igor Mezić. Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control. *Automatica*, 93:149–160, 2018b.

Sanjay Mehrotra. On the implementation of a primal-dual interior point method. *SIAM Journal on Optimization*, 2(4):575–601, 1992.

Robert M. Miura. The Korteweg–deVries equation: A survey of results. *SIAM Review*, 18(3): 412–459, 1976.

Brendan O'Donoghue. Operator splitting for a homogeneous embedding of the linear complementarity problem. *SIAM Journal on Optimization*, 31:1999–2023, 2021.

Joshua L. Proctor, Steven L. Brunton, and J. Nathan Kutz. Generalizing Koopman theory to allow for inputs and control. *SIAM Journal on Applied Dynamical Systems*, 17(1):909–930, 2018.

Bartolomeo Stellato, Goran Banjac, Paul Goulart, Alberto Bemporad, and Stephen Boyd. OSQP: An operator splitting solver for quadratic programs. *Mathematical Programming Computation*, 12(4):637–672, 2020.

Robin Strässer, Manuel Schaller, Julian Berberich, Karl Worthmann, and Frank Allgöwer. Kernel-based error bounds of bilinear Koopman surrogate models for nonlinear data-driven control. *IEEE Control Systems Letters*, 9:1892–1897, 2025.

Dieter Teichrib and Moritz Schulze Darup. Efficient computation of Lipschitz constants for MPC with symmetries. In *62nd IEEE Conference on Decision and Control*, pages 6685–6691, 2023.

Matthew O. Williams, Ioannis G. Kevrekidis, and Clarence W. Rowley. A data-driven approximation of the Koopman operator: Extending dynamic mode decomposition. *Journal of Nonlinear Science*, 25(6):1307–1346, 2015.

Matthew O. Williams, Maziar S. Hemati, Scott T. M. Dawson, Ioannis G. Kevrekidis, and Clarence W. Rowley. Extending data-driven Koopman analysis to actuated systems. *IFAC-PapersOnLine*, 49(18):704–709, 2016.

Liang Wu and Alberto Bemporad. A simple and fast coordinate-descent augmented-Lagrangian solver for model predictive control. *IEEE Transactions on Automatic Control*, 68(11):6860–6866, 2023.

Liang Wu and Richard D. Braatz. A direct optimization algorithm for input-constrained MPC. *IEEE Transactions on Automatic Control*, 70(2):1366–1373, 2025.

Liang Wu and Richard D Braatz. A quadratic programming algorithm with $O(n^3)$ time complexity. *IEEE Transactions on Automatic Control*, pages 1–16, 2026. doi: 10.1109/TAC.2026.3673192.

Andrea Zanelli, Alexander Domahidi, Juan Jerez, and Manfred Morari. FORCES NLP: An efficient implementation of interior-point methods for multistage nonlinear nonconvex programs. *International Journal of Control*, 93(1):13–29, 2020.

Melanie N. Zeilinger, Manfred Morari, and Colin N. Jones. Soft constrained model predictive control with robust stability guarantees. *IEEE Transactions on Automatic Control*, 59(5):1190–1202, 2014.

Xinglong Zhang, Wei Pan, Riccardo Scattolini, Shuyou Yu, and Xin Xu. Robust tube-based model predictive control with Koopman operators. *Automatica*, 137:110114, 2022.

Appendix A. Numerical experiments on general QPs

The proposed BoxQP solver in the dynamics-relaxed KoopmanMPC-to-BoxQP approach can be extended to general QPs, not limited to nonlinear MPC applications. Consider a general convex QP,

$$\min_{x} \ \tfrac{1}{2} x^\top Q x + q^\top x \tag{22a}$$

$$\text{s.t.} \ \ y_{\min} \le Ax \le y_{\max} \tag{22b}$$

$$x_{\min} \le x \le x_{\max} \tag{22c}$$

where $x \in \mathbb{R}^{n_x}$, $Q = Q^\top \succeq 0$ and $Q \in \mathbb{R}^{n_x \times n_x}$, $q \in \mathbb{R}^{n_x}$, and $A \in \mathbb{R}^{n_y \times n_x}$. The bound constraint (22c) often comes from the physical actuator limits, and (22b) comes from user-specified constraints, such as the states/outputs constraints. To ensure feasibility without violating hard actuator limits, the core idea of this article is the use of the soft-constrained formulation as follows,

$$\begin{aligned} &\min_{x,y} \ \tfrac{1}{2} x^\top Q x + q^\top x + \tfrac{\rho}{2}\|Ax - y\|_2^2 = \frac{1}{2}\begin{bmatrix} x \\ y \end{bmatrix}^\top \begin{bmatrix} Q + \rho A^\top A & -\rho A^\top \\ -\rho A & \rho I \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} x \\ y \end{bmatrix}^\top \begin{bmatrix} q \\ 0 \end{bmatrix} \\ &\text{s.t.} \ \begin{bmatrix} x_{\min} \\ y_{\min} \end{bmatrix} \le \begin{bmatrix} x \\ y \end{bmatrix} \le \begin{bmatrix} x_{\max} \\ y_{\max} \end{bmatrix} \end{aligned} \tag{23}$$

where ρ is a large penalty parameter, such as $\rho = 10^6$. The soft-constrained BoxQP (23) is always feasible, Lipschitz-guaranteed, and supports the lightening fast tailored solver (from two reasons: *i)* only requiring roughly 10 iterations for a high-accuracy solution, *ii)* allowing the dimension reduction of linear systems) especially when $n_y \gg n_x$, such as from PDE-MPC applications. Algorithm 1 is compared against OSQP and SCS solvers (all choose the 'eps_rel', '1e-6') on random QPs (22), and Fig. 4 shows that Algorithm 1 is fastest (at least an order of magnitude faster) and only requires roughly 10^{-2} s on QPs with 8400 inequalities.

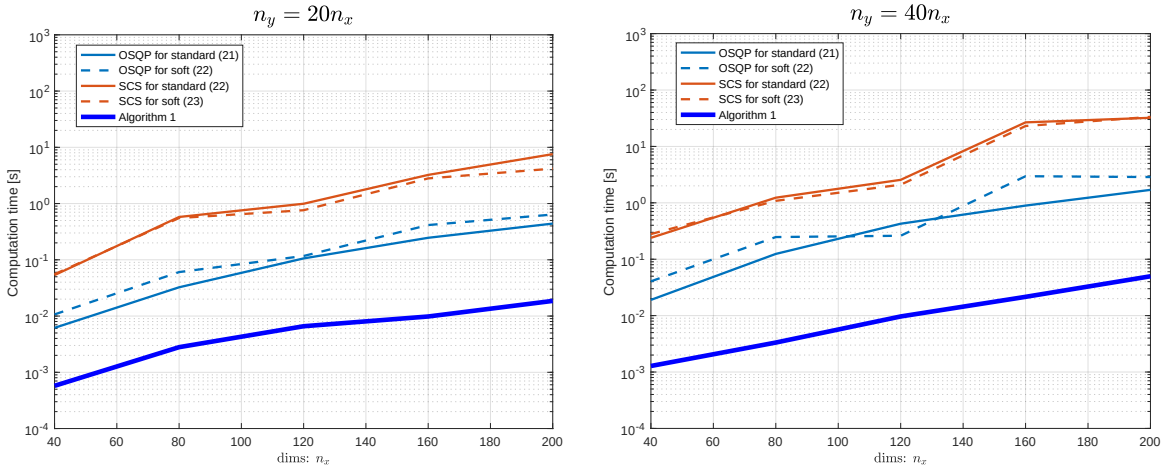

Figure 4: Execution time comparison between Algorithm 1, OSQP, and SCS solvers on random QPs (and their soft-constrained QPs with $\rho = 10^6$) with different dimensions. Left: $n_y = 20n_x$. Right: $n_y = 40n_x$.