Convex Approximations for a Bi-level Formulation of Data-Enabled Predictive Control

Xu Shang
Yang Zheng

Department of Electrical and Computer Engineering, University of California San Diego

Abstract

The Willems’ fundamental lemma, which characterizes linear time-invariant (LTI) systems using input and output trajectories, has found many successful applications. Combining this with receding horizon control leads to a popular Data-Enabled Predictive Control (DeePC) scheme. DeePC is first established for LTI systems and has been extended and applied for practical systems beyond LTI settings. However, the relationship between different DeePC variants, involving regularization and dimension reduction, remains unclear. In this paper, we first discuss a bi-level optimization formulation that combines a data pre-processing step as an inner problem (system identification) and predictive control as an outer problem (online control). We next introduce a series of convex approximations by relaxing some hard constraints in the bi-level optimization as suitable regularization terms, accounting for an implicit identification. These include some existing DeePC variants as well as two new variants, for which we establish their equivalence under appropriate settings. Notably, our analysis reveals a novel variant, called DeePC-SVD-Iter, which has remarkable empirical performance of direct methods on systems beyond deterministic LTI settings.

Keywords: Data-driven Control, Bi-level optimization, Convex approximation

1. Introduction

There has been a surging interest in utilizing data-driven techniques to control systems with unknown dynamics (Pillonetto et al., 2014; Markovsky and Dörfler, 2021). Existing methods can be generally categorized into indirect and direct data-driven control techniques: indirect data-driven control approaches typically include sequential system identification (system ID) (Ljung, 1998; Chiuso and Pillonetto, 2019) and model-based control (Kouvaritakis and Cannon, 2016), while direct data-driven control methods bypass system ID and directly design control strategies from input and output measured data (Markovsky and Dörfler, 2021).

In particular, Data-Enabled Predictive Control (DeePC) (Coulson et al., 2019; Markovsky and Dörfler, 2021) that combines behavioral theory with receding horizon control has received increasing attention. It utilizes Willem’s fundamental lemma (Willems, 2007) to construct a data-driven representation of a dynamic system and incorporates it with receding horizon control. DeePC is first established for deterministic linear time-invariant (LTI) systems, and its equivalence with subspace predictive control (SPC) has been discussed in (Fiedler and Lucia, 2021). Berberich et al. (2020) further investigate conditions for its closed-loop stability. The DeePC approach has shown promising results for the control of practical systems beyond LTI settings (Wang et al., 2023; Elokda et al., 2021; Shang et al., 2023; Lian et al., 2023). For non-deterministic or nonlinear systems, suitable regularizations are necessary for DeePC; see Breschi et al. (2023); Dörfler et al. (2022).

There are different regularization strategies for DeePC, ranging from some heuristics in Coulson et al. (2019) to principled analysis via bi-level formulations in Dörfler et al. (2022). Notably, indirect data-driven control is first formulated as a bi-level optimization problem involving both control and identification in Dörfler et al. (2022). Many regularized versions (such as $l_1$ or $l_2$ norms) of
DeePC can be considered as convex relaxations of the bi-level optimization. Beyond regularization, some recent approaches aim to decrease the optimization dimensions and improve computational efficiency (Zhang et al., 2023; Alsalti et al., 2023). One simple strategy is to use a singular value decomposition (SVD) to pre-process the data-driven representation, which has shown promising performance (Zhang et al., 2023; Yin et al., 2021). However, the relationship between the recent variants of DeePC for non-deterministic and nonlinear systems, involving regularization and dimension reduction, remains unclear, and there is no analysis and comparison for their solution qualities.

In this paper, we introduce a new bi-level formulation incorporating both system ID techniques and predictive control, and discuss how existing and new variants of DeePC can be considered as convex approximations of this bi-level formulation; Figure 1 illustrates the overall process. Specifically, in our bi-level formulation for DeePC, the data pre-processing step is viewed as an inner optimization problem (identification), and the predictive control is viewed as an outer optimization problem (online control). Constraints for the inner optimization problem are derived from system ID methods (e.g., SPC Favoreel et al. (1999), and low-rank approximation Markovsky (2016)). We further discuss a series of convex approximations by relaxing some hard constraints as suitable regularization terms. In this process, we derive two new variants of DeePC by adapting existing methods: 1) Data-Driven-SPC is derived from classical SPC with the same structure as DeePC, and 2) DeePC-SVD-Iter refines the data-driven representation in DeePC-SVD from Zhang et al. (2023) and provides superior performance. We also investigate the equivalence of DeePC-Hybrid (Dörfler et al., 2022), DeePC-SVD (Zhang et al., 2023), and Data-Driven-SPC. Our analysis is more general than Zhang et al. (2023); Fiedler and Lucia (2021); Breschi et al. (2023). Numerical experiments confirm our analysis and show the superior performance of DeePC-SVD-Iter.

The rest of this paper is structured as follows. Section 2 introduces preliminaries and the problem statement of our bi-level formulation. A series of convex approximations are introduced in Section 3, their relationship is established in Section 4. Section 5 compares their control performance via numerical simulations. Finally, we conclude the paper in Section 6. Some technical proofs and extra discussions are provided in our technical report (Shang and Zheng, 2023).

2. Preliminaries and Problem Statement

2.1. Preliminaries

We consider a linear time-invariant (LTI) system in the discrete-time domain:

\[
\begin{align*}
x(k + 1) &= Ax(k) + Bu(k), \\
y(k) &= Cx(k) + Du(k),
\end{align*}
\]

Figure 1: Schematic of data-driven control, which starts by collecting data (usually noisy) from the real system. Indirect methods identify a parametric model, while DeePC forms a Hankel matrix as the trajectory library for predictive control. Our bi-level formulation (7) integrates system ID techniques to DeePC. We introduce a series of convex approximations (8), (9), (10), and (13) that relax the bi-level formulation.
where the state, input, output at time $k$ are $x(k) ∈ \mathbb{R}^n$, $u(k) ∈ \mathbb{R}^m$, and $y(k) ∈ \mathbb{R}^p$, respectively. Given a desired reference trajectory $y_ℓ ∈ \mathbb{R}^{pN}$ with horizon $N > 0$, input constraint set $U ⊆ \mathbb{R}^m$, output constraint set $Y ⊆ \mathbb{R}^p$, we aim to design control inputs such that the system output tracks the reference trajectory. In particular, we consider the well-known receding horizon predictive control

$$
\min_{x, u, y} \sum_{k=t}^{t+N-1} (\|y(k) - y_ℓ(k)\|^2_Q + \|u(k)\|^2_R)
$$

subject to

$$
\begin{align*}
x(k+1) &= Ax(k) + Bu(k), & k ∈ [t, t + N - 1] \\
y(k) &= Cx(k) + Du(k), & k ∈ [t, t + N - 1] \\
x(t) &= x_{ini}, \\
u(k) &∈ U, \ y(k) ∈ Y, & k ∈ [t, t + N - 1],
\end{align*}
$$

where $x_{ini} ∈ \mathbb{R}^n$ is the initial state of system (1) and $\|u(k)\|^2_R$ denotes the quadratic norm $u(k)TRu(k)$ (similarly for $\|·\|_Q$) with $R ∈ \mathbb{S}_+^m$ and $Q ∈ \mathbb{S}_+^p$. We assume $U$ and $Y$ are convex sets. Without loss of generality, we consider a regulation problem (i.e., $y_ℓ(k) = 0$) for the rest of the discussions.

It is clear that (2) is a convex optimization problem (it is indeed a quadratic program when $U$ and $Y$ are polytope), which admits an efficient solution when the model for the system (1) is known, i.e., matrices $A$, $B$, $C$ and $D$ are known. In this work, we focus on the case when the system model and the initial condition $x_{ini}$ are unknown. Instead, we have access to 1) offline data, i.e., a length-$T$ pre-collected input/output trajectory of (1), and 2) online data, i.e., the most recent past input/output sequence of length-$T_{ini}$. Then, (2) can be solved by either indirect system ID and model-based control (Åström and Eykhoff, 1971) or the recent emerging direct data-driven control, such as DeePC and its related approaches (Dörfler et al., 2022; Markovsky and Dörfler, 2021). As discussed in Dörfler et al. (2022), the indirect system ID approach is superior in the case of “variance” noise, while DeePC with suitable regularization terms has better performance in the case of “bias” errors.

2.2. Data-Enabled Predictive Control

We here review the basic setup of DeePC. First, let us introduce a notion of persistent excitation and we use $\text{col}(A_1, A_2, \ldots , A_n) = [A_1^T, A_2^T, \ldots , A_n^T]^T$ in the rest of the paper.

**Definition 1 (Persistently Exciting)** A sequence of signal $ω = \text{col}(ω(1), ω(2), \ldots , ω(T))$ of the length $T$ ($T ∈ \mathbb{N}$) is persistently exciting of order $L$ ($L < T$) if its associated Hankel matrix with depth $L$, defined below, has full row rank,

$$
H_L(ω) = \begin{bmatrix}
ω(1) & ω(2) & \cdots & ω(T-L+1) \\
ω(2) & ω(3) & \cdots & ω(T-L+2) \\
\vdots & \vdots & \ddots & \vdots \\
ω(L) & ω(L+1) & \cdots & ω(T)
\end{bmatrix}.
$$

**Lemma 1 (Fundamental Lemma; Willems et al. (2005))** Suppose that system (1) is controllable. Given a length $T$ input/output trajectory: $u_δ = \text{col}(u_δ(1), \ldots , u_δ(T)) ∈ \mathbb{R}^{mT}$, $y_δ = \text{col}(y_δ(1), \ldots , y_δ(T)) ∈ \mathbb{R}^{pT}$ where $u_δ$ is persistently exciting of order $L + n$, then a length $L$ input/output sequence $(u_s, y_s)$ is a valid trajectory of (1) if and only if there exists a $g ∈ \mathbb{R}^{T-L+1}$ such that

$$
\begin{bmatrix}
H_L(u_δ) \\
H_L(y_δ)
\end{bmatrix} g = \begin{bmatrix} u_s \\
y_s
\end{bmatrix}.
$$

If length $L$ is not smaller than the lag of the system, matrix $\text{col}(H_L(u_δ), H_L(y_δ))$ has rank $mL + n$.  


The DeePC approach in Coulson et al. (2019) employs (3) to build a predictor based on the pre-collected data. In particular, the Hankel matrix formed by the offline data is partitioned as

\[
\begin{bmatrix}
U_p \\
U_f
\end{bmatrix} := \mathcal{H}_L(u_d), \quad \begin{bmatrix}
Y_p \\
Y_f
\end{bmatrix} := \mathcal{H}_L(y_d),
\]

where \( U_p \) and \( U_f \) consist the first \( T_{ini} \) rows and the last \( N \) rows of \( \mathcal{H}_L(u_d) \), respectively (similarly for \( Y_p \) and \( Y_f \); so \( L = T_{ini} + N \)). We denote the most recent past input trajectory of length \( T_{ini} \) and the future input trajectory of length \( N \) as \( u_{ini} = \text{col}(u(t-T_{ini}), u(t-T_{ini}+1), \ldots, u(t-1)) \) and \( u = \text{col}(u(t), u(t+1), \ldots, u(t+N-1)) \), respectively (similarly for \( y_{ini}, y \)).

Then, Lemma 1 ensures that the sequence \( \text{col}(u_{ini}, y_{ini}, u, y) \) is a valid trajectory of (1) if and only if there exists \( g \in \mathbb{R}^{T-T_{ini}-N+1} \) such that (5) holds. For notational simplicity, we further denote the matrix \( \text{col}(U_p, Y_p, U_f, Y_f) \) associated with pre-collected data as \( H \).

Note that \( H \) can be considered as a trajectory library since each of its columns is a valid trajectory of system (1). If \( T_{ini} \) is larger or equal to the lag of system (1), \( y \) is unique given any \( u_{ini}, y_{ini} \) and \( u \) in (5). The most basic version of DeePC (Coulson et al., 2019) utilizes the predictor (5) as the data-driven representation of (2a) to (2c) and reformulate the problem (2) as

\[
\begin{array}{ll}
\min_{y,u,y} & \sum_{k=t}^{t+N-1} \left( \|y(k) - y_r(k)\|_Q^2 + \|u(k)\|_R^2 \right) \\
\text{subject to} & (5), \; u \in \mathcal{U}, \; y \in \mathcal{Y}
\end{array}
\]

where we slightly abuse the notation and use \( u \in \mathcal{U}, y \in \mathcal{Y} \) to denote input/output constraints (2d).

2.3. A bi-level formulation beyond deterministic LTI systems

It is not difficult to show that for LTI systems with noise-free data, problems (2) and (6) are fully equivalent (Coulson et al., 2019, Thm. 5.1). However, for the case beyond deterministic LTI systems, there exist different regularization terms or data pre-processing techniques that extend the basic DeePC (6). Indeed, an extensive discussion on bridging indirect and direct data-driven control was presented in Dörfler et al. (2022), where two different bi-level formulations were discussed.

Motivated by Dörfler et al. (2022), we introduce another bi-level formulation that incorporates data pre-processing techniques from system ID. In practice, the data predictor \( H \) in (5) may be corrupted by “variance” noises and/or “bias” errors. The key idea of our bi-level formulation is to pre-process the raw data \( H \) and construct a new trajectory library \( \tilde{H} := \text{col}(U_p, \tilde{Y}_p, U_f, \tilde{Y}_f) \) satisfying specific system ID:

\[
\begin{align}
\text{minimize} & \quad \sum_{k=t}^{t+N-1} \left( \|y(k) - y_r(k)\|_Q^2 + \|u(k)\|_R^2 \right) + \lambda_y \|\sigma_y\|_2^2 \quad (7a) \\
\text{subject to} & \quad \tilde{H}^* g = \text{col}(u_{ini}, y_{ini} + \sigma_y, u, y), \quad (7b) \\
& \quad \text{where } \tilde{H}^* \in \arg\min_H J(H, \tilde{H}), \quad (7c) \\
& \quad \tilde{Y}_f = Y_f/\text{col}(U_p, \tilde{Y}_p, U_f) \text{ (Row Space)}, \quad (7d) \\
& \quad \text{rank}(\tilde{H}) = mL + n \quad \text{ (Rank Number)}, \quad (7e) \\
& \quad \tilde{H} \in \mathcal{H} \quad \text{ (Hankel Structure)}. \quad (7f)
\end{align}
\]
This bi-level problem structure in (7), which is consistent with those in Dörfler et al. (2022), reflects the sequential ID and control tasks, where we first fit a data-driven model \( \tilde{H} \) from the raw data \( H \) (4) in the inner system ID before using the model for DeePC in the outer problem. The key difference between (7) and the bi-level formulation in Dörfler et al. (2022) is the inner problem in Dörfler et al. (2022) is to identify a parametric model while we directly identify a data-driven model, i.e., we process the data matrix \( H \) via (7d) to (7f).

In the outer problem (7a)-(7b), we have introduced a slack variable \( \sigma_y \) and its regularization term to handle the model mismatch and ensure feasibility, as discussed in Markovsky and Dörfler (2021). In the inner optimization problem (7c)-(7f), \( J(\tilde{H}, H) \) in (7c) denotes system identification loss function with \( H \) being the raw Hankel matrix (4). In (7d), \( Y_F/\text{col}(\tilde{U}_P, \tilde{Y}_P, \tilde{U}_F) \) denotes the orthogonal projection of \( Y_F \) onto the row space of \( \text{col}(\tilde{U}_P, \tilde{Y}_P, \tilde{U}_F) \). This row space constraint is derived from SPC (Favoreel et al., 1999) which will be discussed in detail in Section 3.2. The rank constraint (7e) and Hankel structure (7f) (where \( H \) is the set of all matrices with Hankel structure; cf. Definition 1) come from low-rank approximation in Markovsky (2016). We refer the interested readers to Fiedler and Lucia (2021) and Willems et al. (2005) for further details on row space and rank number respectively.

We will derive a series of convex approximations for this bi-level formulation (7) in Section 3 and discuss their equivalence (if possible) and relationship in Section 4.

### 3. Convex Approximations

While the bi-level formulation (7) is not solvable immediately, it provides useful guidance to derive new formulations/variants of DeePC. In this section, we present four convex approximations by adapting existing methods; see Figure 1 for an overview. These strategies relax the inner constraints (7d) to (7f) using suitable regularizers to the outer problem.

#### 3.1. DeePC with regularization and dimension reduction

We first discuss two existing convex approximations of (7): DeePC-Hybrid from Dörfler et al. (2022) and DeePC-SVD from Zhang et al. (2023). Both of them use two different regularization terms to relax the row space constraint (7d) and the rank constraint (7e) while DeePC-Hybrid keeps the Hankel constraint (7f) and DeePC-SVD drops it.

Compared with the basic DeePC in (6), besides the regularizer \( \|\sigma_y\|_2^2 \), we introduce two extra regularizers \( \|g\|_1 \) and \( \|(I - \Pi_1)g\|_2 \) in DeePC-Hybrid, which reads as

\[
\min_{g, \sigma_y, u \in \mathcal{U}, y \in \mathcal{Y}} \|u\|_R^2 + \|y\|_Q^2 + \lambda_1 \|g\|_1 + \lambda_2 \|(I - \Pi_1)g\|_2^2 + \lambda_y \|\sigma_y\|_2^2
\]

subject to

\[
\begin{bmatrix}
U_P \\
Y_P \\
U_F \\
Y_F \\
\end{bmatrix} g = \begin{bmatrix}
u_{\text{ini}} \\
y_{\text{ini}} + \sigma_y \\
u \\
y \\
\end{bmatrix},
\]

where \( \Pi_1 = H_1^\dagger H_1 \) with \( H_1 = \text{col}(U_P, Y_P, U_F) \). Throughout the rest of the discussion, we denote \( \|u\|_R^2 = \sum_{k=t}^{t+N-1} \|u(k)\|_R^2 \) (similarly for \( \|y\|_Q^2 \)). We note that the \( l_1 \) regularization \( \|g\|_1 \) can be viewed as a convex relaxation of the rank constraint (7e) (Dörfler et al., 2022, Thm. IV.8), while the regularization \( \|(I - \Pi_1)g\|_2^2 \) relaxes row space constraint (7d) (Dörfler et al., 2022, Thm. IV.6).
Since the column number of $H$ is usually larger than its row number in practice (i.e., $H$ is typically a fat matrix), DeePC-SVD in Zhang et al. (2023) utilizes singular value decomposition (SVD) to pre-process $H$ and reduce its column dimension. Denoting the SVD of $H$ as $H = W \Sigma V^T$, where $\Sigma$ contains its non-zero singular values, we construct a new data matrix $\bar{H} = W \Sigma$, and partition its rows as $\bar{H} = \text{col}(\bar{U}_p, \bar{Y}_p, \bar{U}_f, \bar{Y}_f)$. Then, the formulation DeePC-SVD reads as

$$\min_{\bar{g}, \sigma_y, u \in U, y \in Y} \|u\|_R^2 + \|y\|^2_Q + \lambda_1 \|g\|_1 + \lambda_2 \|\bar{H}(I - \bar{\Pi}_1)\bar{g}\|_2^2 + \lambda_y \|\sigma_y\|_2^2$$

subject to

$$\bar{g} = \begin{bmatrix} \bar{U}_p^+ \\ \bar{Y}_p \\ \bar{U}_f \\ \bar{Y}_f \end{bmatrix} u_{\text{ini}} \begin{bmatrix} y_{\text{ini}} + \sigma_y \\ u \\ y \end{bmatrix}$$

where $\bar{\Pi}_1 = \bar{H}_1^T \bar{H}_1$ and $\bar{H}_1 = \text{col}(\bar{U}_p, \bar{Y}_p, \bar{U}_f)$. The dimension of $\bar{g}$ in (9) can be much smaller than that in (8), and this simple fact can improve numerical efficiency.

### 3.2. Data-Driven Subspace Predictive Control (SPC)

We here introduce a new Data-Driven-SPC to approximate (7) and establish its equivalence with the classical SPC in Favoreel et al. (1999). Similar to DeePC-Hybrid (8), we drop the Hankel structure constraint (7f) and use a $l_1$ regularization to relax the rank constraint (7e). However, we will directly handle the row space constraint (7d) without using any relaxation. Let us consider

$$\min_{\bar{H}} \|\text{col}(\bar{U}_p, \bar{Y}_p, \bar{U}_f) - \text{col}(U_p, Y_p, U_f)\|_2$$

subject to $\bar{Y}_F = Y_F/\text{col}(\bar{U}_p, \bar{Y}_p, \bar{U}_f)$.

This inner problem has an analytical solution as $\bar{H}^* = \text{col}(U_p, Y_p, U_f, M)$, with $M = Y_F \Pi_1$ and $\Pi_1$ defined in Section 3.1. Then, we formulate Data-Driven-SPC as a problem in the form of (10).

$$\min_{\sigma_y, u \in U, y \in Y} \|u\|_R^2 + \|y\|^2_Q + \lambda_1 \|g\|_1 + \lambda_y \|\sigma_y\|_2^2$$

subject to

$$g = \begin{bmatrix} U_p^+ \\ Y_p \\ U_f \\ M \end{bmatrix} \begin{bmatrix} u_{\text{ini}} \\ y_{\text{ini}} + \sigma_y \\ u \\ y \end{bmatrix}.$$  

We show that (10) is indeed a direct data-driven version of (11) in the sense that they produce the same solution under a very mild condition. The proof is in our report (Shang and Zheng, 2023).

**Theorem 1**  If $Q > 0, R > 0, \lambda_1 = 0$ and $H_1 = \text{col}(U_p, Y_p, U_f)$ has full row rank, then (10) and (11) have the same optimal solution $u^*, y^*, \sigma_y^*, \forall \lambda_y > 0$.

Note that our new Data-Driven-SPC (10) is more flexible than the classical SPC (11) thanks to the parameter $\lambda_1$, which was motivated from the relaxation of the rank constraint (7e).

### 3.3. DeePC with dominant range space and Hankel structure

All the convex approximations (8), (9) and (10) use different regularizations to relax the difficult constraints (7d) and (7e), but both (9) and (10) directly drop the Hankel constraint (7f). In this subsection, we derive another new convex approximation for (7) that also approximates the Hankel structure with a dominant range space from the SVD. We call it as DeePC-SVD-Iter.
In particular, we consider (12) as the inner problem in the bi-level formulation (7), where the row space constraint (7d) will be relaxed using regularization. Note that (12) is also difficult to solve due to the interplay between (12b) and (12c). There are extensive results in the field of structured low-rank approximation (SLRA) (Markovsky, 2008). One idea is to use an alternative optimization strategy by considering (12b) and (12c) sequentially. Specifically, we here adapt an iterative SLRA algorithm in Yin and Smith (2021) to get an approximation solution to (12). We first note that problem (12) without (12c) admits an analytical solution $\hat{H}^* = W_r \Sigma_r V_r^T$ where $W_r, \Sigma_r$ and $V_r$ represent the leading $mL + n$ singular vectors and singular values of $H$, i.e., the dominant rank space.

The key idea of the iterative SLRA is to utilize SVD for low-rank approximation of noisy data and then project the low-rank matrix to the set of Hankel matrices. This process is summarized in Algorithm 1. For notational simplicity, we define $H_u = H_L(u_d)$, $H_y = H_L(y_d)$ to denote the Hankel matrices in (4). Thanks to the persistent excitation on $u_d$ with no noise, we have rank($H_u) = mL$. However, the measurement $y_d$ usually contain “variance” noise and “bias” error, thus the data matrix satisfies rank(col($H_u, H_y)) > mL + n$. We use an iterative procedure to denoise $H_y$ while maintaining its Hankel structure (note that we do not change $H_u$ since it is normally noisy-free).

Let $\Pi_2 = H_u^T H_u$ be the orthogonal projector onto the row space of $H_u$, and we first compute $H_y(I - \Pi_2)$ which is the component of $H_y$ in the null space of $H_u$. In each iteration of Algorithm 1, we perform an SVD of $H_y(I - \Pi_2)$, estimate its rank-$n$ approximation (since we have rank($H_u) = mL$), and finally combine it with the component of $H_y$ in the row space of $H_u$ as follows

$$H_y(I - \Pi_2) = \sum_{i=1}^{nL} \sigma_i u_i v_i^T, \quad \hat{H}(H_y) := H_y \Pi_2 + \sum_{i=1}^{n} \sigma_i u_i v_i^T.$$  

We then project $\hat{H}(H_y)$ onto the set of Hankel matrices by averaging skew-diagonal elements and denote $\Pi_H$ as the corresponding operator. The resulting matrix $H_y^*$ from Algorithm 1 is partitioned as col$(Y_p^*, Y_F^*)$, and we form a new Hankel matrix $\hat{H}^* = \text{col}(\hat{U}_p, Y_p^*, \hat{U}_F, Y_F^*)$. Finally, we perform an SVD of $\hat{H}^* = \hat{W}_r \Sigma_r \hat{V}_r^T$ to reduce its column dimension and set $\hat{W}_r \Sigma_r = \text{col}(\hat{U}_p, \hat{Y}_p, \hat{U}_F, \hat{Y}_F)$ with rank $mL + n$ (in the final predictor, we only have the dominant $mL + n$ singular values). This new matrix col$(\hat{U}_p, \hat{Y}_p, \hat{U}_F, \hat{Y}_F)$ is used as the predictor in DeePC as

$$\begin{align*}
\min_{\hat{g}, \sigma_y, u, y} & \quad \|u\|_R^2 + \|y\|_Q^2 + \lambda_2 \|\hat{g}\|_2^2 + \lambda_y \|\sigma_y\|_2^2 \\
\text{subject to} & \quad \begin{bmatrix} \hat{U}_p & \hat{Y}_p & \hat{U}_F & \hat{Y}_F \end{bmatrix} \hat{g} = \begin{bmatrix} u_{\text{ini}} \\ y_{\text{ini}} + \sigma_y \\ u \\ y \end{bmatrix} \\
& \quad \hat{H}_1 = \hat{H}_1^T \hat{H}_1, \quad \tilde{H}_1 = \text{col}(\hat{U}_p, \hat{Y}_p, \hat{U}_F) \quad \text{and} \quad \|\hat{g}\|_2^2 \quad \text{is the relaxation term from the row space constraint. We call this formulation (13) as DeePC-SVD-Iter. In our experiments, Algorithm 1 terminated in a finite number of iterates (up to 2500) with $\epsilon$ chosen from $10^{-3}$ to $10^{-6}$.
\end{align*}$$

**Algorithm 1: Iterative SLRA**

**Input:** $H_y, I_2, n, \epsilon$

$H_{y_1} \leftarrow H_y$

**repeat**

$H_{y_2} \leftarrow \hat{H}(H_{y_1})$ SVD step

$H_{y_1} \leftarrow H_{y_1} \Pi_H(H_{y_2})$ Hankel proj

**until** $\|H_{y_1} - H_{y_2}\| \leq \epsilon \|H_{y_1}\|$;

**Output:** $H_y^* = H_{y_2}$
4. Relationship among Different Convex Approximations

As motivated above, (8) to (10) and (13) are all tractable convex approximations for the bi-level formulation (7). They all begin with the same data matrix $H$ and apply different relaxation strategies to deal with the identification constraints (7d) to (7f). We here look into their relationships and establish certain equivalence. Proofs of Theorems 2 and 3 are presented in Shang and Zheng (2023).

First, it is not difficult to see that all of them are equivalent when the data matrix $H$ comes from an LTI system with no noise. We summarize this simple fact below.

**Proposition 1** Suppose that the data matrix $H$ in (4) comes from a controllable LTI system (1) with no noise, and the input $u_d$ is persistently exciting of order $L + n$. Let $\lambda_1 = 0$, $\lambda_2 = 0$ and $\sigma_y = 0$. Then, all DeePC variants in (8), (9), (10), and (13) have the same unique optimal solution $u^*, y^*$. For a controllable system (1) with no noise, the data matrix $H$ has already satisfied row space (7d), rank number (7e), and Hankel structure constraints (7f). Then the new matrix $H^*$ after pre-processing in Data-Driven-SPC (10) and DeePC-SVD-Iter (13) will remain the same as that in DeePC-Hybrid (8), and their range space are also equal to that of DeePC-SVD (9). Thus, all these formulations have the same feasible region and cost functions, and they are equivalent.

We next move away from noise-free LTI systems. The data matrix $H$ may have “variance” noise and/or “bias” errors; see Dorfler et al. (2022). In this case, we can still show that DeePC-Hybrid (8) and DeePC-SVD (9) produce the same optimal solution $u^*, y^*, \sigma_y^*$. We establish Theorem 2 by expressing $g$ and $\bar{g}$ in terms of $u, y, \sigma_y$. Then, using the SVD properties, we show that (8) and (9) become strictly convex optimization problems with the same objective function, decision variables, and feasible region. Note that Theorem 2 includes Theorem 1 of Zhang et al. (2023) as a special case, where Zhang et al. (2023) requires $U$ and $Y$ are convex polytopes that allow simple KKT conditions in their proof.

Finally, DeePC-Hybrid, DeePC-SVD and Data-Driven-SPC are also equivalent under certain conditions. This is summarized in Theorem 3 below.

**Theorem 2** Fix any data matrix $H$, and suppose $\lambda_1 = 0$, $U$ and $Y$ are convex. Then, DeePC-Hybrid (8) and DeePC-SVD (9) have the same optimal solution $u^*, y^*, \sigma_y^*$, $\forall \lambda_2 > 0, \lambda_y > 0$.

We establish Theorem 2 by expressing $g$ and $\bar{g}$ in terms of $u, y, \sigma_y$. Then, using the SVD properties, we show that (8) and (9) become strictly convex optimization problems with the same objective function, decision variables, and feasible region. Note that Theorem 2 includes Theorem 1 of Zhang et al. (2023) as a special case, where Zhang et al. (2023) requires $U$ and $Y$ are convex polytopes that allow simple KKT conditions in their proof.

Finally, DeePC-Hybrid, DeePC-SVD and Data-Driven-SPC are also equivalent under certain conditions. This is summarized in Theorem 3 below.

**Theorem 3** Fix any data matrix $H$, and suppose $\lambda_1 = 0$, $\lambda_y > 0$ and $U$ and $Y$ are convex sets. If $\lambda_2$ is sufficiently large, DeePC-Hybrid (8) and DeePC-SVD (9) have the same unique optimal solution $u^*, y^*$ and $\sigma_y^*$ as Data-Driven-SPC (10).

The key idea in the proof is to transform the regularizer $\lambda_2 \|I - \Pi_1\|_2^2$ in (8) as the constraint $\|I - \Pi_1\|_2 = 0$ when $\lambda_2$ is sufficiently large via penalty arguments. Then, (8) and (10) have the same objective function and decision variables. The proof is completed by further establishing that they have the same feasible region. From Theorems 1 to 3, we conclude that Data-Driven-SPC (10), DeePC-Hybrid (8), DeePC-SVD (9) are equivalent to classical SPC (11) with noisy data when $\lambda_1 = 0$, $\lambda_y > 0$, $\lambda_2$ is sufficiently large and $H_1$ has full row rank. This is more general than Fiedler and Lucia (2021); Breschi et al. (2023): the equivalence for DeePC-Hybrid with regularizer $\|g\|_2^2$ and classical SPC is discussed in Fiedler and Lucia (2021), while DeePC-Hybrid and an approach similar to Data-Driven-SPC are proved to be equivalent in Breschi et al. (2023).

Note that the new variant DeePC-SVD-Iter involves an iterative algorithm to pre-process the noisy data (Algorithm 1), and thus it is non-trivial to formally establish its relationship with respect to other variants. Yet, our numerical experiments in Section 5 show that DeePC-SVD-Iter often has superior performance among all these convex approximations for noisy data.
Figure 2: Equivalent optimal solutions of different convex approximations in Theorems 2 and 3. (a) All methods with noise-free data. (b) DeePC-Hybrid and DeePC-SVD with $\lambda_1 = 0, \lambda_2 = 30$ and $\lambda_y = 100$. (c) DeePC-Hybrid, DeePC-SVD and Data-Driven-SPC with $\lambda_1 = 0, \lambda_2 = 10000$ and $\lambda_y = 100$.

5. Numerical Experiments

We perform numerical experiments to illustrate Theorems 2 and 3\(^1\). We also numerically investigate the effects of $\lambda_1$, $\lambda_2$, and confirm the superior performance of DeePC-SVD-iter (13). Additional numerical results on nonlinear systems are provided in our report (Shang and Zheng, 2023).

Experiment setup. We consider an LTI system from Fiedler and Lucia (2021). This is a triple-mass-spring system with $n = 8$ states, $m = 2$ inputs (two stepper motors), and $p = 3$ outputs (disc angles). In our experiments, the length of the pre-collected trajectory is $T = 200$, and the prediction horizon and the initial sequence are chosen as $N = 40$ and $T_{\text{ini}} = 4$, respectively. We choose $Q = I$, $R = 0.I$ and $U = [-0.7, 0.7]$.

Equivalence. We here numerically verify that the optimal solutions from different convex approximations are the same under appropriate settings. For noise-free pre-collected data (Proposition 1), all methods have the same optimal solution, and one solution instance is given in Figure 2(a). We next consider data collection with additive Gaussian measurement noises $\omega \sim \mathcal{N}(0, 0.01I)$. We choose $\lambda_1 = 0, \lambda_2 = 30$ and $\lambda_y = 100$ according to Theorem 2. One solution instance is shown in Figure 2(b), which shows that DeePC-Hybrid and DeePC-SVD provide the same optimal solution. Finally, for the equivalence of Data-Driven-SPC, DeePC-Hybrid and DeePC-SVD, we choose $\lambda_1 = 0$ and $\lambda_2 = 10000$ according to Theorem 3, and the results are shown in Figure 2(c).

Influence of $\lambda_1$ and $\lambda_2$. We then analyze the effect of hyperparameters $\lambda_1$ and $\lambda_2$. In particular, similar to Dörfler et al. (2022), we consider the realized control cost after applying the optimal control inputs from (8) to (10) and (13), which is computed as $\|u_{\text{opt}}\|^2_R + \|y_{\text{true}}\|^2_Q$, where $u_{\text{opt}}$ is the computed optimal control input and $y_{\text{true}}$ is the realized trajectory after applying $u_{\text{opt}}$. We fixed $\lambda_y = 100$.

Figure 3 shows the realized control performance over $\lambda_1$ and $\lambda_2$ for different convex approximations. The hyperparameters $\lambda_1$ and $\lambda_2$ indeed have a significant effect for DeePC-Hybrid (8), DeePC-SVD (9) and Data-Driven-SPC (10) (denoted as DDSPC in Figure 3). For these methods, $\lambda_1$ needs to be chosen more carefully, neither too large nor too small and $\lambda_2$ should not be chosen too small; similar phenomena also appeared.

---

\(^1\) Our code is available at https://github.com/soc-ucsd/Convex-Approximation-for-DeePC.
Table 1: Realized Control Cost and Computational Time; GT denotes ground truth with noisy-free data.

<table>
<thead>
<tr>
<th></th>
<th>GT</th>
<th>Hybrid</th>
<th>SVD</th>
<th>Data-Driven-SPC</th>
<th>SVD-Iter</th>
<th>System ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Realized Cost</td>
<td>277.25</td>
<td>388.42</td>
<td>370.20</td>
<td>365.08</td>
<td>288.17</td>
<td>279.86</td>
</tr>
<tr>
<td>Increasing Rate</td>
<td>N/A</td>
<td>40.1%</td>
<td>33.5%</td>
<td>31.7%</td>
<td>3.9%</td>
<td>0.9%</td>
</tr>
<tr>
<td>Compu. time [s]</td>
<td>N/A</td>
<td>0.133</td>
<td>0.131</td>
<td>0.097</td>
<td>0.104</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Figure 4: Open-loop trajectories (the angle of disc 2) of different methods. The blue trajectory and orange trajectories represent ground truth and different approximation methods, respectively.

in Dörfler et al. (2022). However, it is notable that DeePC-SVD-Iter (13) not only achieves the best performance but also is not very sensitive to $\lambda_2$ (note that $\lambda_1 = 0$ in (13)).

Comparison of DeePC variants. Finally, we compare the realized control cost and the computational time for different convex approximations. Motivated by Figure 3, we choose $\lambda_1 = \lambda_3 = 30$ and $\lambda_y = 100$. The performance of DeePC variants is related to the pre-collected trajectory. Thus, all presented realized control costs and computational time for different convex approximations are averaged over 100 pre-collected trajectories. The numerical results are listed in Table 1. The ground-truth cost is computed from (6) with noise-free data and we use the subspace approach N4SID proposed in Van Overschee and De Moor (1994) to compute the cost for System ID. From Table 1, we see that the realized control cost satisfies DeePC-Hybrid > DeePC-SVD > Data-Driven-SPC > DeePC-SVD-Iter > System ID. For the LTI system with noisy data, the inner problem in (7) forces the data-driven representation to be more structured, which enhances noise rejection performance for upper predictive control in (7). The increasing rate of realized cost for our new DeePC-SVD-Iter is 3.9%, which is much better than other DeePC variants.

Figure 4 shows one typical open-loop trajectory for all methods. In this case, the open-loop trajectories from (8) to (10) and (13) remain close to the ground truth up to 2 s. Then the trajectory is better aligned with the ground truth from Fig. 4(a) to Fig. 4(e) as the corresponding data-driven representation becomes more structured. Our numerical results also suggest that the indirect system ID approach is superior in the case of “variance” noise, consistent with Dörfler et al. (2022). In the report Shang and Zheng (2023), our numerical results on nonlinear systems further reveal that DeePC-SVD-Iter (13) also has enhanced performance in the case of “bias” errors.

6. Conclusion

In this paper, we have proposed a new bi-level formulation incorporating system ID techniques and predictive control. The existing DeePC (i.e., DeePC-Hybrid (8) and DeePC-SVD (9)) and also new variants (i.e., Data-driven-SPC (10) and DeePC-SVD-Iter (13)) can be considered as convex approximations of this bi-level formulation. We have further clarified their equivalence under appropriated settings (Theorems 1 to 3). Numerical simulations have validated our theoretical findings, and also revealed the superior performance of DeePC-SVD-Iter (13) with a more structured predictor. Interesting future directions include analyzing the effect of the length of pre-collected data, and investigating the closed-loop performance of different DeePC variants.
Acknowledgments

This work is supported in part by NSF ECCS-2154650, NSF CMMI-2320697, NSF CAREER-2340713, and an Early Career Faculty Development Award from the Jacobs School of Engineering at UC San Diego.

References


