Toward Near-Globally Optimal Nonlinear Model Predictive Control via Diffusion Models

Tzu-Yuan Huang[1*] TZU-YUAN.HUANG@TUM.DE
Armin Lederer [2*] ARMIN.LEDERER@INF.ETHZ.CH
Nicolas Hoischen[1] NICOLAS.HOISCHEN@TUM.DE
Jan Brüdigam[1] JAN.BRUEDIGAM@TUM.DE
Xuehua Xiao[1] XUEHUA.XIAO@TUM.DE
Stefan Sosnowski [1] SOSNOWSKI@TUM.DE
Sandra Hirche[1] HIRCHE@TUM.DE
[1]*Chair of Information-oriented Control, Technical University of Munich, Germany*
[2]*Learning and Adaptive Systems Group, Department of Computer Science, ETH Zurich, Switzerland*
[*]*Equal contribution*

Editors: N. Ozay, L. Balzano, D. Panagou, A. Abate

Abstract

Achieving global optimality in nonlinear model predictive control (NMPC) is challenging due to the non-convex nature of the underlying optimization problem. Since commonly employed local optimization techniques depend on carefully chosen initial guesses, this non-convexity often leads to suboptimal performance resulting from local optima. To overcome this limitation, we propose a novel diffusion model-based approach for near-globally optimal NMPC consisting of an offline and an online phase. The offline phase employs a local optimizer to sample from the distribution of optimal NMPC control sequences along generated system trajectories through random initial guesses. Subsequently, the generated diverse dataset is used to train a diffusion model to reflect the multi-modal distribution of optima. In the online phase, the trained model is leveraged to efficiently perform a variant of random shooting optimization to obtain near-globally optimal control sequences without relying on any initial guesses or online NMPC solving. The effectiveness of our approach is illustrated in a numerical simulation indicating high performance benefits compared to direct neural network approximations of NMPC and significantly lower computation times than online solving NMPC using global optimizers.

Keywords: Diffusion model, approximate model predictive control, global optimization

1. Introduction

Nonlinear Model predictive control (NMPC) (Rawlings et al., 2017) is a modern control technique that iterates between solving an optimal control problem and applying the first element of the solution's control sequence to the nonlinear system. Due to the ease of considering constraints and the flexibility for optimizing performance metrics, NMPC has immense success in a large variety of applications ranging from robot manipulators (Heins and Schoellig, 2023) and unmanned aerial vehicles (Lindqvist et al., 2020) to chemical engineering (Drgoňa et al., 2017).

The optimal control problems inherent in NMPC are generally non-convex, but commonly solved using gradient-based optimization techniques in practice (Rao et al., 1998; Qin and Badgwell, 1997). However, such methods only ensure convergence to a local optimum close to the initial guess. The optimized cost function can reflect metrics such as energy consumption (Oldewurtel et al., 2012), agricultural output (Ding et al., 2018), and material consumption (Ghezzi et al., 2023).

Therefore, poor local optima are suboptimal solutions that can have an immediate detrimental real-world effect. This has led to the development of globally optimal NMPC approaches, often relying on randomness to ensure sufficient exploration, e.g., particle swarm optimization (Xu et al., 2016) and the cross-entropy method (Wang and Ba, 2020; Pinneri et al., 2021). However, these approaches are computationally demanding, restricting their application to settings with low sampling times.

Since NMPC is already considered computationally expensive without the integration of global optimization, different approaches have been proposed to reduce its online complexity. A straightforward approach is to reduce the number of iterations required by the numerical solver using warm starts (Yildirim and Wright, 2002), which can be obtained from a function approximator trained from offline generated solutions (Celestini et al., 2024) or policies obtained via reinforcement learning (Hart et al., 2019; Reiter et al., 2024). While this approach still requires online optimization, explicit NMPC approaches aim at directly determining a function that solves the optimal control problem of NMPC (Bemporad et al., 2002). In recent years, it has become increasingly popular to achieve this by using flexible function approximators such as neural networks to learn the NMPC control law in a supervised fashion from numerically solved optimal control problems (Hertneck et al., 2018; Karg and Lucia, 2020; Nubert et al., 2020; Bonzanini et al., 2021; Tagliabue and How, 2024). Even though these approaches significantly reduce the complexity of NMPC, they do not consider the challenges of the multi-extremal optimization landscapes.

This issue has recently received increasing attention in the related field of trajectory planning and off-policy reinforcement learning, where generative models have shown promising results (Urain et al., 2024). In particular, diffusion models are highly effective at learning multi-modal representations from sample data (Huang et al., 2023). For trajectory planning, this can be leveraged to employ complex trajectory optimization techniques to generate expert data, whose distribution can be learned with a diffusion model to enable fast online planning (Janner et al., 2022; Carvalho et al., 2023). In offline reinforcement learning, diffusion models conditioned on normalized rewards can learn to predict the best state sequence from roll-outs of arbitrary policies jointly with a tracking controller (Ajay et al., 2023). Moreover, the full action distribution can be directly learned from demonstrations of the policy (Chi et al., 2023), which can be used to warm start model predictive control (Li et al., 2024) and for efficient sampling-based online optimization (Zhou et al., 2024). However, these approaches focus on different challenges compared to the problem of learning a globally optimal NMPC controller: High diversity of samples is the major goal of diffusion models for planning, while in offline reinforcement learning the training data of diffusion models cannot be influenced.

In this paper, we address the problem of finding a tractable approximation of globally optimal NMPC via diffusion models. For this purpose, we empirically approximate a multi-modal distribution over the optimal control sequences of NMPC in the form of a data set, which we obtain using local optimization methods that are optimized for low computation times. This distribution can be straightforwardly learned from the dataset using diffusion models, such that new samples can be drawn from it with comparatively little computational complexity. Finally, we exploit the model knowledge to employ a variant of the random shooting method to find control sequences close to the global optimum using samples from the diffusion model. The numerical results suggest that our approach based on learning a distribution over solutions strongly outperforms methods directly learning the control law while exhibiting lower complexity than an online global optimization strategy.

2. Problem Formulation

We consider a nonlinear discrete-time control system defined by differentiable system dynamics

$\boldsymbol{f} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$, i.e.,

$$\boldsymbol{x}_{t+1} = \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t), \tag{1}$$

with time $t \in \mathbb{N}$, state vector $\boldsymbol{x}_t \in \mathbb{X} \subset \mathbb{R}^{n_x}$ in the compact set $\mathbb{X}$, and control input $\boldsymbol{u}_t \in \mathbb{U} \subset \mathbb{R}^{n_u}$. We assume that the inputs $\boldsymbol{u}_t$ are constrained to some known compact set $\mathcal{U} \subset \mathbb{U}$. This work aims to find an optimal input sequence $\boldsymbol{u}^*_{[0:H]}(\boldsymbol{x}_t) = \left[\boldsymbol{u}^*_0(\boldsymbol{x}_t), \ldots, \boldsymbol{u}^*_{H-1}(\boldsymbol{x}_t)\right]$ over a time horizon $H \in \mathbb{N}$, solving the following Optimal Control Problem (OCP) at every time step t

$$\min_{\boldsymbol{u}_{[0:H]}} J(\boldsymbol{x}_t, \boldsymbol{u}_{[0:H]}) := \sum_{i=0}^{H-1} \ell(\hat{\boldsymbol{x}}_i, \boldsymbol{u}_i) + V_f(\hat{\boldsymbol{x}}_H) \tag{2a}$$

$$\text{s.t.} \quad \hat{\boldsymbol{x}}_{i+1} = \boldsymbol{f}(\hat{\boldsymbol{x}}_i, \boldsymbol{u}_i), \ \hat{\boldsymbol{x}}_0 = \boldsymbol{x}_t, \tag{2b}$$

$$\boldsymbol{u}_{[0:H]} = \left[\boldsymbol{u}_0, \ldots, \boldsymbol{u}_{H-1}\right], \ \boldsymbol{u}_i \in \mathcal{U}, \ i = 0, \ldots, H-1. \tag{2c}$$

where the cost function $J : \mathbb{X} \times \mathbb{U}^H \to \mathbb{R}$ consists of the differentiable stage cost $\ell : \mathbb{X} \times \mathbb{U} \to \mathbb{R}$ and the differentiable terminal cost $V_f : \mathbb{X} \to \mathbb{R}$. In the NMPC framework, given an initial state $\boldsymbol{x}_0$, the OCP (2) can be solved recursively over a moving horizon $[t, t+H]$ at each discrete sampling time instant t. This process generates an optimal input sequence $\boldsymbol{u}^*_{[0:H]}(\boldsymbol{x}_t)$ at every time step t, and only the first element $\boldsymbol{u}^*_0(\boldsymbol{x}_t)$ in this sequence is applied as the control input of the current time step.

While (2) generally constitutes a non-convex optimization problem for nonlinear dynamics (1), it is commonly solved numerically using gradient-based optimization techniques. Considering these numerical solvers as black boxes, they can be interpreted as operators $\mathbb{S} : \mathcal{C}^1(\mathbb{U}^H) \times \mathbb{U}^H \to \mathcal{U}^H$ that map the cost function $J(\boldsymbol{x}_t, \cdot)$ to a locally optimal solution $\hat{\boldsymbol{u}}_{[0:H]}(\boldsymbol{x}_t)$ which depends on the initial guess $\tilde{\boldsymbol{u}}_{[0:H]}$. Even though one of the locally optimal solutions $\hat{\boldsymbol{u}}_{[0:H]}(\boldsymbol{x}_t)$ corresponds to the globally optimal solution $\boldsymbol{u}^*_{[0:H]}(\boldsymbol{x}_t)$, we generally do not know which initial guess $\tilde{\boldsymbol{u}}_{[0:H]}$ needs to be chosen to obtain it. Therefore, we cannot expect to find the globally optimal control sequence $\boldsymbol{u}^*_{[0:H]}(\boldsymbol{x}_t)$ without further assumptions on the initial guess $\tilde{\boldsymbol{u}}_{[0:H]}$, such that we consider a slightly weaker notion of optimality in the following, which we refer to as probabilistic near-global optimality. We define a stochastic policy $\boldsymbol{\pi} : \mathbb{R}^{n_u} \times \mathbb{R}^{n_x} \to \mathbb{R}_{\geq 0}$ to satisfy this form of optimality, if

$$P(||\boldsymbol{u} - \boldsymbol{u}^*_{[0:H]}(\boldsymbol{x}_t)||_2 \leq \varepsilon) \geq 1 - \delta \text{ for } \boldsymbol{u} \sim \boldsymbol{\pi}(\cdot|\boldsymbol{x}_t) \tag{3}$$

for constants $\varepsilon, \delta \in \mathbb{R}_{\geq 0}$. The goal of this paper is the development of a method for finding such probabilistic near-globally optimal control laws $\boldsymbol{\pi}$ satisfying (3) with arbitrarily small ε, δ without relying on local online optimization.

Remark 1 *Note that we do not consider state constraints in the OCP* (2) *because our focus lies in efficiently finding approximations of the globally optimal control sequence* $\boldsymbol{u}^*_{[0:H]}(\boldsymbol{x}_t)$. *Since the stochastic policy* (3) *does not inherit exact constraint satisfaction from the OCP* (2) *in general, extending it to state constraints is left for future work.*

3. Approximating Nonlinear Model Predictive Control via Diffusion Models

We propose an approach for approximating globally optimal NMPC via diffusion models based on a procedure with an offline training phase and an online optimization phase. In the offline training phase, we generate samples from a distribution over the optimal solutions of the NMPC problem (2) by exploiting computationally efficient local optimization methods. The details of our employed sampling scheme are explained in Section 3.1. These samples are used to train a diffusion model in Section 3.2, which can approximate the distribution of optimal control sequences arbitrarily accurately given sufficient model flexibility and data set size. Therefore, generating a sufficiently large number of samples using the diffusion model will yield at least one control sequence close to the globally optimal one, which is exploited for efficient online optimization in Section 3.3.

3.1. Data Generation for Diffusion-Approximated NMPC

While the multi-extremal nature of NMPC's optimization landscape is well-known, it is not trivial to describe the corresponding solutions. Aligning this representation with our goal of training a diffusion model, we propose to use a probability distribution $\rho(\cdot|\boldsymbol{x}_t) : \mathbb{U}^H \to \mathbb{R}$ over the control sequences, which peaks at the solutions of (2). This distribution can be straightforwardly defined using the solution operator $\mathbb{S}$ and an uniformative, flat distribution $\phi : \mathbb{U}^H \to \mathbb{R}$ with support on $\mathcal{U}^H$, i.e.,

$$\mathbb{S}(J(\boldsymbol{x}_t, \cdot), \tilde{\boldsymbol{u}}_{[0:H]}) \sim \rho(\cdot|\boldsymbol{x}_t) \qquad\qquad \text{for } \tilde{\boldsymbol{u}}_{[0:H]} \sim \phi. \tag{4}$$

Even though determining an explicit description of $\rho(\cdot|\boldsymbol{x}_t)$ is generally intractable, sampling from it merely requires sampling from ϕ and subsequently solving the OCP (2) numerically. This can be efficiently realized using local optimization methods designed for fast optimization in NMPC. Therefore, the distribution $\rho(\cdot|\boldsymbol{x}_t)$ can be effectively represented through samples in principle.

As $\rho(\cdot|\boldsymbol{x}_t)$ is a conditional distribution dependent on the current state $\boldsymbol{x}_t$, it does not suffice to generate these samples for a single state. However, densely covering $\mathbb{X}$, e.g., by using a uniform distribution over $\mathbb{X}$, is also intractable for many problems due to the curse of dimensionality. Therefore, we design a distribution that puts an emphasis on states close to the trajectories generated under the locally optimal policies starting from a region of interest. This distribution relies on roll-outs of control inputs sampled from $\rho(\cdot|\boldsymbol{x}_t)$ starting from initial states $\boldsymbol{x}_0 \sim \chi$ randomly drawn from an initial distribution $\chi : \mathbb{X} \to \mathbb{R}$, which is only supported on a comparatively small subset $\mathbb{X}_0 \subset \mathbb{X}$. Since the employed policy in this procedure does generally not correspond exactly to the policy that will be used for the control of the actual system, this approach will experience the covariance shift phenomenon leading to compounding errors between trajectories in the data set and trajectories generated by the policy applied to the system (Rajaraman et al., 2020). To mitigate this issue, we augment the sampling scheme by randomly perturbing states with Gaussian noise $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\sigma}(t))$ with time-dependent variance $\boldsymbol{\sigma}^2 : \mathbb{N} \to \mathbb{R}_{\geq 0}$. Thereby, we generate a data set that will asymptotically be dense in $\mathbb{X}$ and thus, inherits robustness against compounding errors from the uniform distribution, while exhibiting significant advantages for learning policies in the areas of interest defined via trajectories. The complete algorithm for sampling states and optimal sequences to generate a representative data set $\mathcal{D}$ of $\rho(\cdot|\boldsymbol{x}_t)$ is summarized in Algorithm 1.

It is important to note that we only sample one control sequence $\hat{\boldsymbol{u}}_{[0:H]}(\boldsymbol{x}_t)$ per state $\boldsymbol{x}_t$ from the distribution of solutions $\rho(\cdot|\boldsymbol{x}_t)$, which merely requires the execution of a local optimization method to evaluate $\mathbb{S}$. This is in contrast to multistart techniques for global optimization that require multiple samples from $\rho(\cdot|\boldsymbol{x}_t)$ (Danilova et al., 2022). However, it is straightforward to show that this approach asymptotically leads to a data set with that contains a dense subset of global optima.[1]

Theorem 2 *Consider data sets $\mathcal{D}$ generated using Algorithm 1 with positive $\boldsymbol{\sigma}$. If $P(\boldsymbol{u}^*_{[0:H]}(\boldsymbol{x})) > 0$ for all $\boldsymbol{x} \in \mathbb{X}$ under the distribution induced by ϕ and $\mathbb{S}$, then, for every $\boldsymbol{x} \in \mathbb{X}$, $\epsilon \in \mathbb{R}$, $\bar{\delta} \in \mathbb{R}$, there exist $N_s, N_p, N_T \in \mathbb{N}$ such that $(\boldsymbol{x}', \boldsymbol{u}^*_{[0:H]}(\boldsymbol{x}')) \in \mathcal{D}$ and $||\boldsymbol{x} - \boldsymbol{x}'|| \leq \epsilon$ with probability $1 - \bar{\delta}$.*

The assumption of positive probabilities $P(\boldsymbol{u}^*_{[0:H]}(\boldsymbol{x}))$ is necessary to ensure that global optimum can be found at all, but it does not pose a severe restriction in practice. In fact, this requirement can be easily satisfied using an initial guess distribution ϕ with positive support on a set $\mathbb{U}^H \supset \mathcal{U}^H$.

[1] Proofs for all results can be found in the extended version (https://mediatum.ub.tum.de/doc/1779727).

Algorithm 1 Optimizer-based Data Generation

```
Require: NMPC solver $\mathbb{S}$, perturbation variance $\boldsymbol{\sigma}$, initial guess distribution $\phi$, initial state distribution $\chi$, cost function $J$, sample numbers $N_s, N_T, N_p \in \mathbb{N}$
for $j = 0, 1, \cdots, N_s$ do
    Sample initial state $\boldsymbol{x}_0 \sim \chi$
    for $t = 0, 1, \cdots, N_T$ do
        for $i = 1, \ldots, N_p$ do
            Generate perturbed state $\boldsymbol{x}_t^d \leftarrow \boldsymbol{x}_t + \boldsymbol{\epsilon}$, $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}(t))$
            Sample initial guess $\tilde{\boldsymbol{u}}_{[0:H]} \sim \phi$
            Execute local optimizer $\hat{\boldsymbol{u}}_{[0:H]}(\boldsymbol{x}_t^d) \leftarrow \mathbb{S}(J(\boldsymbol{x}_t^d, \cdot), \tilde{\boldsymbol{u}}_{[0:H]})$
            Augment data set $\mathcal{D} \leftarrow \mathcal{D} \cup \{\boldsymbol{x}_t^d, \hat{\boldsymbol{u}}_{[0:H]}(\boldsymbol{x}_t^d)\}$
        Apply $\hat{\boldsymbol{u}}_0(\boldsymbol{x}_t)$ to system (1): $\boldsymbol{x}_{t+1} \leftarrow \boldsymbol{f}(\boldsymbol{x}_t^d, \hat{\boldsymbol{u}}_0(\boldsymbol{x}_t^d))$
```

Since the Gaussian perturbations of states in Algorithm 1 additionally induce a state distribution with support on $\mathbb{X}$, it immediately follows that Algorithm 1 is guaranteed to provide a suitable coverage of the globally optimal control sequences $\boldsymbol{u}^*_{[0:H]}(\boldsymbol{x}_t)$ when generating sufficiently large data sets $\mathcal{D}$. Thereby, it increases the robustness against compounding errors without the necessity to train multiple policies as in on-policy imitation learning (Ross et al., 2011). This is a crucial difference as the training of diffusion models exhibits a considerably higher computational complexity compared to shallow neural networks, which are the main focus of on-policy imitation learning.

3.2. Diffusion Model Training for Approximating NMPC

The conditional distribution $\rho(\cdot|\boldsymbol{x}_t)$, which is used to generate the data set $\mathcal{D}$ in Section 3.1, is potentially multi-modal due to the possible existence of multiple optima in the NMPC problem (2). To learn a representation of this distribution, we leverage diffusion models due to their recent success in learning high-dimensional distributions from sample data (Sohl-Dickstein et al., 2015; Ho et al., 2020). The core idea behind diffusion models is an iterative denoising procedure, which is learned through an iterative noising process. Starting from the noiseless control sequence sample $\hat{\boldsymbol{u}}_{[0:H]}(\boldsymbol{x}_t)$ collected by Algorithm 1, the noising process is based on iteratively adding Gaussian noise. The goal of this noising process is to perturb the distribution of optimal control sequences to Gaussian noise in the limit of infinite noising steps. More formally, this forward process, also known as the diffusion process, can be described by

$$q(\boldsymbol{u}^k|\boldsymbol{u}^{k-1}) := \mathcal{N}(\boldsymbol{u}^k; \sqrt{1-\beta_k}\boldsymbol{u}^{k-1}, \beta_k\mathbf{I}), \quad 1 \leq k \leq K \in \mathbb{N}, \quad \boldsymbol{u}^0 = \hat{\boldsymbol{u}}_{[0:H]}(\boldsymbol{x}_t), \tag{5}$$

where $0 < \beta_1, \cdots, \beta_K < 1$ are scheduled variances. In terms of noise variables, this process is equivalently described using coefficients $\alpha_k := \prod_{j=1}^k (1-\beta_j)$, which results in

$$\boldsymbol{u}^k = \sqrt{\alpha_k}\boldsymbol{u}^0 + \sqrt{(1-\alpha_k)}\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{6}$$

Due to this forward process, the distribution of optimal control sequences can be described through a reverse process

$$p_{\boldsymbol{\theta}}(\hat{\boldsymbol{u}}_{[0:H]}(\boldsymbol{x}_t)) = \int p_{\boldsymbol{\theta}}(\boldsymbol{u}^{0:K}) d\boldsymbol{u}^{1:K} = \int p(\boldsymbol{u}^K) \prod_{k=1}^{K} p_{\boldsymbol{\theta}}(\boldsymbol{u}^{k-1}|\boldsymbol{u}^k) d\boldsymbol{u}^{1:K} \tag{7}$$

where the joint distribution $p_{\boldsymbol{\theta}}(\boldsymbol{u}^{0:K})$ is defined as a Markov chain with Gaussian transitions

$$p_{\boldsymbol{\theta}}(\boldsymbol{u}^{k-1}|\boldsymbol{u}^k) := \mathcal{N}(\boldsymbol{u}^{k-1}; \boldsymbol{\mu}_{\boldsymbol{\theta}}(\boldsymbol{u}^k, k), \boldsymbol{\Sigma}_k) \tag{8}$$

Algorithm 2 NMPC via Diffusion Models

```
Require: diffusion model $p_{\boldsymbol{\theta}}(\cdot|\cdot)$, number of optimization samples $M$
while true do
    Measure system state $\boldsymbol{x}_t$
    for $m = 1, \cdots, M$ do
        Sample control sequence $\hat{\boldsymbol{u}}^m_{[0:H]} \sim p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}_t)$
        Compute cost value $v^m \leftarrow J(\boldsymbol{x}_t, \hat{\boldsymbol{u}}^m_{[0:H]})$
    Obtain control sequence $m^* \leftarrow \arg\min_{m=1,\ldots,M} v^m$
    Apply control input $\boldsymbol{x}_{t+1} \leftarrow \boldsymbol{f}(\boldsymbol{x}_t, \hat{\boldsymbol{u}}^{m^*}_0)$, $t \leftarrow t+1$
```

starting from a standard normal distribution $p(\boldsymbol{u}^K)$. In this reverse process, the mean functions $\boldsymbol{\mu}_{\boldsymbol{\theta}}$ of the conditional distributions $p_{\boldsymbol{\theta}}(\boldsymbol{u}^{k-1}|\boldsymbol{u}^k)$ can be represented by a neural network with parameters $\boldsymbol{\theta}$, which are optimized to represent the data distribution $\rho(\cdot|\boldsymbol{x}_t)$. When parameterized via $\boldsymbol{\mu}_{\boldsymbol{\theta}}(\boldsymbol{u}^k, k) = a_k \boldsymbol{u}^k + b_k \boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{u}^k, k)$ with $a_k = {}^1/\sqrt{1-\beta_k}$ and $b_k = {}^{-\beta_k}/(\sqrt{1-\beta_k}\sqrt{1-\alpha_k})$, training the parameters $\boldsymbol{\theta}$ can be effectively performed by minimizing the loss $\mathbb{E}_{k\in[1,K],\boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\mathbf{I})}[||\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{u}^k, k)||^2]$, with the predicted noise $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{u}^k, k)$. Hence, we essentially train a neural network $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}$ using the value $\boldsymbol{u}^k$ defined in the forward process (6) to obtain the mean $\boldsymbol{\mu}_{\boldsymbol{\theta}}$ in the conditional distributions (8).

While diffusion models have been originally developed to learn a fixed distribution, it is straightforward to extend them to learning state-dependent distributions as needed for control. For this purpose, we apply the classifier-free diffusion guidance (Ho and Salimans, 2022) to learn a conditional diffusion model. In this method, the noise predictor integrates both conditional information $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{u}^k, k, \boldsymbol{x}_t)$ and unconditional information $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{u}^k, k, \varnothing)$ during training, where $\varnothing$ is a null token for the condition $\boldsymbol{x}_t$. Therefore, it employs a training loss

$$\mathcal{L} = \mathbb{E}_{k\in[1,K],\boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\mathbf{I}),\boldsymbol{x}_t\sim\mathcal{D},b\sim\text{Bernoulli}(p)}[||\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{u}^k, k, (1-b)\cdot\boldsymbol{x}_t + b\cdot\varnothing)||_2^2], \tag{9}$$

with a probability $p \in [0, 1]$ for the unconditioned information. Since diffusion models are capable of capturing arbitrary data distributions given sufficiently flexible model architectures $p_{\boldsymbol{\theta}}$ (Sohl-Dickstein et al., 2015), this approach enables us to learn an arbitrarily accurate approximation $p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}_t)$ of the distribution $\rho(\cdot|\boldsymbol{x}_t)$ over optimal control sequences $\hat{\boldsymbol{u}}_{[0:H]}(\boldsymbol{x}_t)$ given a sufficiently large data set $\mathcal{D}$. Hence, for every $\boldsymbol{x} \in \mathbb{X}$ and $\boldsymbol{u}_{[0:H]} \in \mathbb{U}^H$, we can achieve

$$\left| \int_{\boldsymbol{v}\in\mathbb{B}_\varepsilon(\boldsymbol{u}_{[0:H]})} \rho(\boldsymbol{v}|\boldsymbol{x}_t) - p_{\boldsymbol{\theta}}(\boldsymbol{v}|\boldsymbol{x}_t)\mathrm{d}\boldsymbol{v} \right| \leq \tilde{\delta}, \tag{10}$$

where $\mathbb{B}_\varepsilon(\boldsymbol{u}_{[0:H]})$ is a ball with radius ε and center $\boldsymbol{u}_{[0:H]}$, and $\tilde{\delta} \in (0, 1]$ is a probability depending on $\bar{\delta}$ in Theorem 2.

3.3. Toward Near-Globally Optimal NMPC using Diffusion Models

The diffusion model provides us with a representation $p_{\boldsymbol{\theta}}$ of the probability distribution of optimal control sequences ρ, which does not involve the application of the numerical solver $\mathbb{S}$. Sampling from $p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}_t)$ is particularly straighforward: After sampling an initial control sequence $\boldsymbol{u}^K \sim \mathcal{N}(0, \boldsymbol{I})$, we iteratively sample $\boldsymbol{u}^{k-1} \sim p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{u}^k)$ from the conditional distributions (8) until we obtain $\hat{\boldsymbol{u}}^m_{[0:H]} = \boldsymbol{u}^0$. Therefore, we can sample $\hat{\boldsymbol{u}}^m_{[0:H]} \sim p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}_t)$ through an iterative procedure.

Since we can ensure that the diffusion model exhibits a low approximation error, i.e., $p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}) \approx \rho(\cdot|\boldsymbol{x})$, this possibility for approximately sampling from the distribution of optimal control sequences $\hat{\boldsymbol{u}}_{[0:H]}(\boldsymbol{x}_t)$ provides us with a computationally efficient approach toward near-globally optimal NMPC as outlined in Algorithm 2. Instead of executing a numerical optimizer, we perform a random shooting approach by sampling multiple control sequences $\hat{\boldsymbol{u}}^m_{[0:H]} \sim p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{x}_t)$, $m = 1, \ldots, M$ with $M \in \mathbb{N}$. For each sample $\hat{\boldsymbol{u}}^m_{[0:H]}$, we compute the corresponding cost $v^m = J(\boldsymbol{x}_t, \hat{\boldsymbol{u}}^m_{[0:H]})$, such that a simple comparison allows to select the best control sequence $\boldsymbol{u}^{m^*}_{[0:H]}$, where $m^* = \arg\min_{m=1,\ldots,M} v^m$. The result $\boldsymbol{u}^{m^*}_{[0:H]}$ of this sample, score, and rank approach is a random variable, whose distance to the globally optimal control sequence $\boldsymbol{u}^*_{[0:H]}(\boldsymbol{x}_t)$ can be probabilistically bounded to ensure near-global optimality as defined in (3).

Theorem 3 *Consider a diffusion model satisfying* (10) *with parameters* $\tilde{\delta}, \varepsilon$ *for all* $\boldsymbol{x} \in \mathbb{X}$. *If* $\tilde{\delta} < p_{\mathbb{B}} = \int_{\boldsymbol{v} \in \mathbb{B}_\varepsilon(\boldsymbol{u}_{[0:H]}(\boldsymbol{x}_t))} \rho(\boldsymbol{v}|\boldsymbol{x}_t) \mathrm{d}\boldsymbol{v}$ *for all* $\boldsymbol{x} \in \mathbb{X}$ *and balls* $\mathbb{B}_\varepsilon(\boldsymbol{u}_{[0:H]}(\boldsymbol{x}_t))$ *with radius* ε *and center* $\boldsymbol{u}_{[0:H]}(\boldsymbol{x}_t)$*, then, the policy defined by Algorithm 2 is probabilistically near-globally optimal, i.e., it satisfies* (3)*, with constants* ε *and* $\delta = (p_{\mathbb{B}} - \tilde{\delta})^M$.

This result guarantees that we can find an arbitrarily close approximation of the globally optimal policy with high probability, with the particular quality of the approximation accuracy depending on the number of generated training samples via Theorem 2 and the expressivity of the employed diffusion model. This guarantee straightforwardly follows from a positive probability that a sample falls within an ε-ball around the global optimum for sufficiently high model accuracy, i.e., $\tilde{\delta} < p_{\mathbb{B}}$. Since the samples are independent, we can multiply these probabilities to bound the probability δ that none of the samples is close to $\boldsymbol{u}^*_{[0:H]}(\boldsymbol{x}_t)$, which yields $\delta = (p_{\mathbb{B}} - \tilde{\delta})^M$. Hence, δ can be made arbitrarily small by choosing a sufficiently large M, such that computational online complexity and accuracy can be effectively balanced.

This trade-off is particularly beneficial since sampling from the diffusion model is highly parallelizable. While we can also make use of parallelization for global optimization using multiple initializations of local optimizers, existing fast solver implementations such as ACADOS (Verschueren et al., 2021) and FATROP (Vanroye et al., 2023) are restricted to CPUs. This severely limits the number of OCPs that can be solved in parallel in practice. In contrast, diffusion models are designed for an evaluation on GPUs whose success is based on parallelization, such that the degree of parallelization can be orders of magnitude higher. Thereby, our diffusion model-based approach allows the generation of larger numbers of samples, which implies control sequences closer to the global optimum due to Theorem 3.

4. Numerical Evaluation

In this section, we evaluate our proposed framework to answer the following questions: (1) Can our approach learn an effective representation of the optima of OCP solved for NMPC? (2) Does it provide computational and performance advantages compared to baselines?

4.1. Simulation Setting

We apply our approach to the following dynamic systems: a) cart-pole, b) pendubot, and c) double cart-pole in our evaluation. The initial condition for each dynamics is such that the pole rests in a neighborhood of the downward position. The control task is to perform a swing-up to the unstable upward position. For this purpose, we define quadratic cost functions for each system, which are summed up over horizons of 64 (a), 256 (b), and 128 (c) time steps. Initial states are sampled from a uniform distribution defined over a neighborhood of the stable downward position. The control task for these systems is well-suited for the investigation of our approach since we know that with

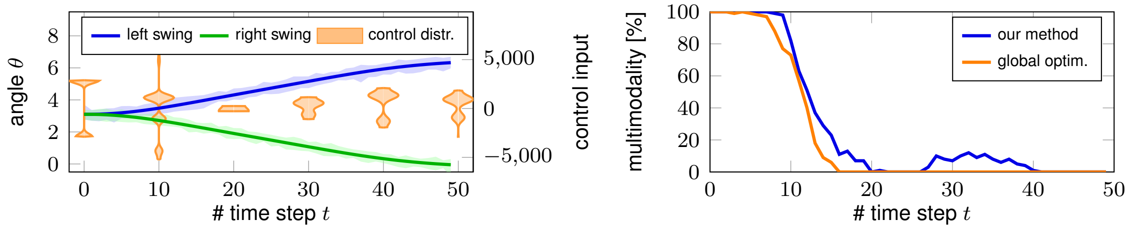

Figure 1: Left: Evolution of the pole angle θ resulting from two kinds of optimal control sequences, whose multimodal distribution is illustrated for different time steps. Right: Comparison of the multimodality of samples from our trained diffusion model and samples employed by a global optimizer for different time steps along roll-out trajectories.

the left and right swing up, generally, at least two locally optimal solutions of the OCP exist. As illustrated on the left side of Fig. 1 for the initial state distribution of the cart-pole system, this leads to a multi-modal distribution of the NMPC solutions in general.

For all the dynamic systems, Algorithm 1 is used to generate the data set for the representation of the multi-modal distribution with the sampling time $0.01s$. The initial guesses are sampled from a uniform distribution ϕ, and the range of ϕ is adjusted to $[-\bar{u}_{t-1}, \bar{u}_{t-1}], t = 1, 2, \cdots, N_T$, where $\bar{u}_0 \in \mathbb{U}$. The numerical optimization is executed using CasADi (Andersson et al., 2019) with an interior-point optimizer (IPOPT) (Wächter and Biegler, 2006). The diffusion model is trained using the loss (9) with probability $p = 0.25$ for the unconditioned information. More details regarding our simulation setup can be found in the extended version of this paper at `https://mediatum.ub.tum.de/doc/1779727`.

We compare our diffusion model approach for NMPC to the following baseline methods:

- **Locally Optimal Model Predictive Control**: We use CasADI (Andersson et al., 2019) with IPOPT (Wächter and Biegler, 2006) as a baseline of a common local optimization-based NMPC method. The results from previous time steps are used to warm-start the optimization.
- **Globally Optimal Model Predictive Control**: As a global optimization NMPC approach, we use CasADI (Andersson et al., 2019) with mulitple random initializations. Thereby, this approach also relies on a sample, rank, and score approach, but requires execution of the local optimizer for sampling.
- **Neural Network Approximation of Model Predictive Control (NN)**: We use the data set generated by Algorithm 1 to train a neural network model of MPC in a supervised fashion as an example of behavior cloning suggested, e.g., by Hertneck et al. (2018). The neural network has 3 hidden layers, each comprised of 50 neurons with $\tanh$ activation functions.
- **Neural Network Approximation of Globally- Optimal Model Predictive Control (NN*)**: We sample multiple random initial guesses to create a data set of approximately globally optimal control sequences, which are used to train a neural network approximation of globally optimal NMPC in a supervised fashion. To ensure a fair comparison, we keep the overall number of solved OCPs constant by setting reduced values of N_s, N_T, and N_p in Algorithm 1.

All the algorithms are implemented in Python, and the simulations are conducted on a machine with an AMD EPYC 7542 32 cores processor and NVIDIA Tesla P100. Note that we refrain from using optimized implementations of NMPC solvers such as CASADOS (Verschueren et al., 2021) or FATROP (Vanroye et al., 2023) since our diffusion model implementation is also not exploiting techniques for accelerated code execution.

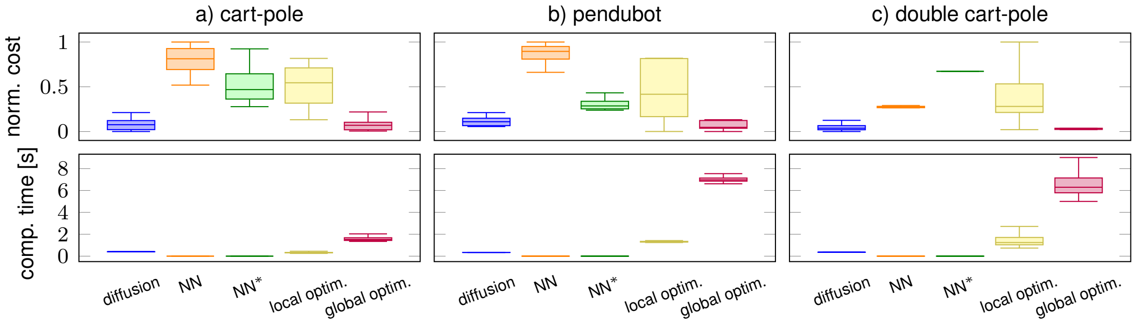

Figure 2: Top: Our diffusion model-based approach achieves similar cost as a global optimization approach for NMPC, but significantly outperforms direct neural network approximations of NMPC. Bottom: The required computation times for online optimization based on a diffusion model are significantly reduced compared to the direct global optimization approach and at most on par with the commonly employed local optimizer.

4.2. Result in Simulation

We first investigate question (1) regarding the capability of our proposed diffusion model approach for learning a multimodal representation of the optimal control sequences. To answer this question, we take the cart-pole system as an example and compare our approach to NMPC with multiple initial guesses as introduced in Section 4.1. Starting from 100 initial states, we compute the roll-out trajectories of both control laws for 50 time steps. For each state $\boldsymbol{x}_t$, $M = 20$ samples are drawn from $\rho(\cdot|\boldsymbol{x}_t)$ through numerical optimization or via the diffusion model. Therefore, we can measure the multi-modality of the distributions, i.e., if multiple optima have been found, by determining if the distance between any of the 20 control sequences exceeds a threshold. The resulting percentages for finding multiple optima are illustrated on the right side of Fig. 1, which clearly demonstrates that our diffusion model is capable of achieving the same diversity as obtained with randomized optimization in NMPC. Note that the loss of multi-modality at later time steps is due to the nature of the considered control problem: once a swing-up in one direction has been executed for a certain number of time steps, the local minimum corresponding to the other swing-up direction becomes extremely large, such that it eventually disappears. Therefore, our diffusion model approach for NMPC is well-suited to reflect the distribution of optimal control sequences for the cartpole system.

Based on this insight, we compare our diffusion model approach for NMPC with baseline methods to investigate its advantages, thereby answering the question (2). For this, we sample initial states from χ randomly for each method and generate roll-outs for the swing-up control task. For each method, we determine the cost experienced during the closed-loop control roll-outs and measure the computation times needed for obtaining the control inputs. The results of this comparison are depicted in Fig. 2. It can be clearly seen that our diffusion model approach for NMPC achieves a lower cost than all baseline methods in this scenario except for the NMPC approach with multiple initial guesses, which performs similarly. Since this approach directly employs global optimization, it poses a lower bound on the achievable cost underlining the strength of our method. It is worth noting that neural network-based approximations of NMPC benefit from lower computational times due to fast forward inference, but struggle to capture the multimodal nature of the underlying control distributions and consequently exhibit inferior performance. Furthermore, approaches requiring globally optimized training targets suffer from limited data availability, negatively impacting gener-

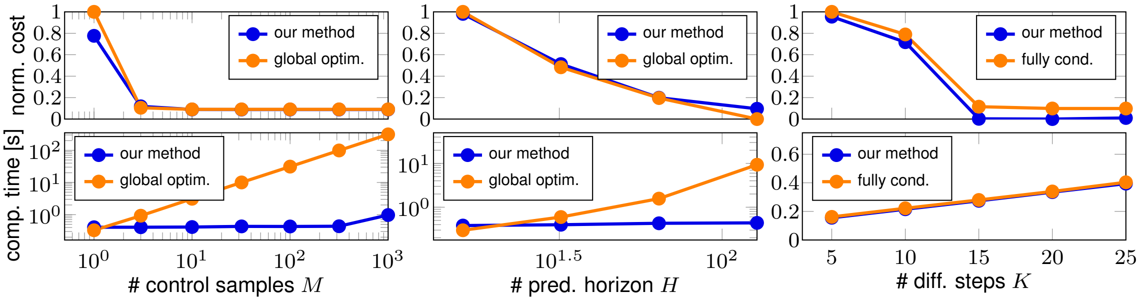

Figure 3: Performance of diffusion model-based and global optimization-based NMPC depending on the number of sampled control sequences M (left), horizon length H (middle), and diffusion steps K (right).

alization. While our diffusion model approach is a significant improvement over the neural network approximation of the globally optimal NMPC scheme for all the investigated system dynamics, its complexity is comparable to or lower than the locally optimal NMPC method. We emphasize that our diffusion model has not been optimized for fast inference, such that significantly lower complexity can be expected using improved sampling schemes (Song et al., 2020; Frans et al., 2024).

While advantages of our diffusion model approach can be observed for the considered exemplary systems, they only exhibit a modest complexity. However, the computational benefits of the diffusion model become significantly more pronounced when a more challenging optimization landscape requires many more control sequence samples M or when longer prediction horizons H are considered, as illustrated on the left and center of Fig. 3, respectively. While the performance of our diffusion model approach to NMPC closely reflects the performance of the globally optimal MPC scheme for the cart-pole system over a range of parameters M and H, the computation time barely changes. In contrast, a steep increase in the computation time is visible for the globally optimal MPC method due to the increasing number of optimizations and growing complexity of each optimization, respectively. Finally, our diffusion model approach is also robust against parameter choices as illustrated on the right of Fig. 3. The number of diffusion steps K only affects the performance when chosen below a threshold $K < 15$, while a fully conditional model with $p = 0$ in the training loss (9) merely causes a small negative effect on the performance. Therefore, this evaluation demonstrates the potential of our diffusion model approach to reliably enable probabilistically, near-globally optimal MPC for control problems beyond the complexity considered in this paper.

5. Conclusion

In this paper, we propose a diffusion model approach to approximate globally optimal nonlinear model predictive control. After training a diffusion model with locally optimal control sequences offline, our approach employs the diffusion model for efficient sampling-based optimization in the online phase. The effectiveness of our diffusion model approach toward near-globally optimal nonlinear model predictive control is demonstrated on the cartpole system. Future work will focus on deriving formal theoretical optimality guarantees for our proposed approach. Moreover, we will investigate the applicability of diffusion models for model predictive control of complex systems, such that we can evaluate it in real-world experiments.

Acknowledgments

This work was supported by the European Union's Horizon Europe innovation action programme under grant agreement No. 101093822, "SeaClear2.0", the DAAD programme Konrad Zuse Schools of Excellence in Artificial Intelligence, Consolidator Grant "Safe data-driven control for human-centric systems" (CO-MAN) of the European Research Council (ERC) and by a part of NCCR Automation, a National Centre of Competence (or Excellence) in Research, funded by the Swiss National Science Foundation (grant number 51NF40 225155)

References

Anurag Ajay, Yilun Du, Abhi Gupta, Joshua B. Tenenbaum, Tommi S. Jaakkola, and Pulkit Agrawal. Is conditional generative modeling all you need for decision making? In *Proceedings of the International Conference on Learning Representations*, 2023.

Joel AE Andersson, Joris Gillis, Greg Horn, James B Rawlings, and Moritz Diehl. Casadi: a software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, 11:1–36, 2019.

Alberto Bemporad, Manfred Morari, Vivek Dua, and Efstratios N Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20, 2002.

Angelo D. Bonzanini, Joel A. Paulson, Georgios Makrygiorgos, and Ali Mesbah. Fast approximate learning-based multistage nonlinear model predictive control using gaussian processes and deep neural networks. *Computers & Chemical Engineering*, 145:107174, 2021.

Joao Carvalho, An T Le, Mark Baierl, Dorothea Koert, and Jan Peters. Motion planning diffusion: Learning and planning of robot motions with diffusion models. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1916–1923, 2023.

Davide Celestini, Daniele Gammelli, Tommaso Guffanti, Simone D'Amico, Elisa Capello, and Marco Pavone. Transformer-based model predictive control: Trajectory optimization via sequence modeling. *IEEE Robotics and Automation Letters*, 9(11):9820–9827, 2024.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, page 02783649241273668, 2023.

Marina Danilova, Pavel Dvurechensky, Alexander Gasnikov, Eduard Gorbunov, Sergey Guminov, Dmitry Kamzolov, and Innokentiy Shibaev. Recent theoretical advances in non-convex optimization. In *High-Dimensional Optimization and Probability: With a View Towards Data Science*, pages 79–163. Springer, 2022.

Ying Ding, Liang Wang, Yongwei Li, and Daoliang Li. Model predictive control and its application in agriculture: A review. *Computers and Electronics in Agriculture*, 151:104–117, 2018.

Ján Drgoňa, Martin Klaučo, Filip Janeček, and Michal Kvasnica. Optimal control of a laboratory binary distillation column via regionless explicit mpc. *Computers & Chemical Engineering*, 96: 139–148, 2017.

Kevin Frans, Danijar Hafner, Sergey Levine, and Pieter Abbeel. One step diffusion via shortcut models. *arXiv preprint arXiv:2410.12557*, 2024.

Andrea Ghezzi, Florian Messerer, Jacopo Balocco, Vincenzo Manzoni, and Moritz Diehl. An implicit and explicit dual model predictive control formulation for a steel recycling process. *European Journal of Control*, 74:100841, 2023.

Patrick Hart, Leonard Rychly, and Alois Knoll. Lane-merging using policy-based reinforcement learning and post-optimization. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 3176–3181. IEEE, 2019.

Adam Heins and Angela P Schoellig. Keep it upright: Model predictive control for nonprehensile object transportation with obstacle avoidance on a mobile manipulator. *IEEE Robotics and Automation Letters*, 2023.

Michael Hertneck, Johannes Köhler, Sebastian Trimpe, and Frank Allgöwer. Learning an approximate model predictive controller with guarantees. *IEEE Control Systems Letters*, 2(3):543–548, 2018.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851, 2020.

Siyuan Huang, Zan Wang, Puhao Li, Baoxiong Jia, Tengyu Liu, Yixin Zhu, Wei Liang, and Song-Chun Zhu. Diffusion-based generation, optimization, and planning in 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16750–16761, 2023.

Michael Janner, Yilun Du, Joshua B Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. *International Conference on Machine Learning*, 2022.

Benjamin Karg and Sergio Lucia. Efficient representation and approximation of model predictive control laws via deep learning. *IEEE Transactions on Cybernetics*, 50(9):3866–3878, 2020.

Anjian Li, Zihan Ding, Adji Bousso Dieng, and Ryne Beeson. Efficient and guaranteed-safe non-convex trajectory optimization with constrained diffusion model. *arXiv preprint arXiv:2403.05571*, 2024.

Björn Lindqvist, Sina Sharif Mansouri, Ali-akbar Agha-mohammadi, and George Nikolakopoulos. Nonlinear mpc for collision avoidance and control of uavs with dynamic obstacles. *IEEE Robotics and Automation Letters*, 5(4):6001–6008, 2020.

Julian Nubert, Johannes Köhler, Vincent Berenz, Frank Allgöwer, and Sebastian Trimpe. Safe and fast tracking on a robot manipulator: Robust mpc and neural network control. *IEEE Robotics and Automation Letters*, 5(2):3050–3057, 2020.

Frauke Oldewurtel, Alessandra Parisio, Colin N. Jones, Dimitrios Gyalistras, Markus Gwerder, Vanessa Stauch, Beat Lehmann, and Manfred Morari. Use of model predictive control and weather forecasts for energy efficient building climate control. *Energy and Buildings*, 45:15–27, 2012.

Cristina Pinneri, Shambhuraj Sawant, Sebastian Blaes, Jan Achterhold, Joerg Stueckler, Michal Rolinek, and Georg Martius. Sample-efficient cross-entropy method for real-time planning. In *Conference on Robot Learning*, pages 1049–1065. PMLR, 2021.

S Joe Qin and Thomas A Badgwell. An overview of industrial model predictive control technology. In *AIche symposium series*, volume 93, pages 232–256, 1997.

Nived Rajaraman, Lin Yang, Jiantao Jiao, and Kannan Ramchandran. Toward the fundamental limits of imitation learning. *Advances in Neural Information Processing Systems*, 33:2914–2924, 2020.

Christopher V Rao, Stephen J Wright, and James B Rawlings. Application of interior-point methods to model predictive control. *Journal of optimization theory and applications*, 99:723–757, 1998.

James Blake Rawlings, David Q Mayne, Moritz Diehl, et al. *Model predictive control: theory, computation, and design*, volume 2. Nob Hill Publishing Madison, WI, 2017.

Rudolf Reiter, Andrea Ghezzi, Katrin BaumgÃĪrtner, Jasper Hoffmann, Robert D McAllister, and Moritz Diehl. Ac4mpc: Actor-critic reinforcement learning for nonlinear model predictive control. *arXiv preprint arXiv:2406.03995*, 2024.

Stephane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15, pages 627–635. PMLR, 11–13 Apr 2011.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, pages 2256–2265. PMLR, 07–09 Jul 2015.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

Andrea Tagliabue and Jonathan P How. Efficient deep learning of robust policies from mpc using imitation and tube-guided data augmentation. *IEEE Transactions on Robotics*, 2024.

Julen Urain, Ajay Mandlekar, Yilun Du, Mahi Shafiullah, Danfei Xu, Katerina Fragkiadaki, Georgia Chalvatzaki, and Jan Peters. Deep generative models in robotics: A survey on learning from multimodal demonstrations. *arXiv preprint arXiv:2408.04380*, 2024.

Lander Vanroye, Ajay Sathya, Joris De Schutter, and Wilm Decré. Fatrop: A fast constrained optimal control problem solver for robot trajectory optimization and control. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10036–10043. IEEE, 2023.

Robin Verschueren, Gianluca Frison, Dimitris Kouzoupis, Jonathan Frey, Niels van Duijkeren, Andrea Zanelli, Branimir Novoselnik, Thivaharan Albin, Rien Quirynen, and Moritz Diehl. acados – a modular open-source framework for fast embedded optimal control. *Mathematical Programming Computation*, 2021.

Tingwu Wang and Jimmy Ba. Exploring model-based planning with policy networks. *International Conference on Learning Representations*, 2020.

A. Wächter and L. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Program*, 106:25–27, 2006.

Fang Xu, Hong Chen, Xun Gong, and Qin Mei. Fast nonlinear model predictive control on fpga using particle swarm optimization. *IEEE Transactions on Industrial Electronics*, 63(1):310–321, 2016.

E Alper Yildirim and Stephen J Wright. Warm-start strategies in interior-point methods for linear programming. *SIAM Journal on Optimization*, 12(3):782–810, 2002.

Guangyao Zhou, Sivaramakrishnan Swaminathan, Rajkumar Vasudeva Raju, J Swaroop Guntupalli, Wolfgang Lehrach, Joseph Ortiz, Antoine Dedieu, Miguel Lázaro-Gredilla, and Kevin Murphy. Diffusion model predictive control. *arXiv preprint arXiv:2410.05364*, 2024.