Nonlinear Model Predictive Control Based on Constraint-Aware Particle Filtering/Smoothing

Iman Askari[1], Shen Zeng[2] and Huazhen Fang[1]

Abstract— **Nonlinear model predictive control (NMPC) has gained widespread use in many applications. Its formulation traditionally involves repetitively solving a nonlinear constrained optimization problem online. In this paper, we investigate NMPC through the lens of Bayesian estimation and highlight that the Monte Carlo sampling method can offer a favorable way to implement NMPC. We develop a constraint-aware particle filtering/smoothing method and exploit it to implement NMPC. The new sampling-based NMPC algorithm can be executed easily and efficiently even for complex nonlinear systems, while potentially mitigating the issues of computational complexity and local minima faced by numerical optimization in conventional studies. The effectiveness of the proposed algorithm is evaluated through a simulation study.**

I. INTRODUCTION

Nonlinear model predictive control (NMPC) has emerged as one of the most important control methods, with numerous applications ranging from process control and robotics to energy management and traffic control [1]–[3]. Its immense success is owed to its ability of predictively optimizing the control of a nonlinear system subject to constraints. At the center of its formulation, NMPC is designed to address an optimal control problem repetitively at every time instant in a receding-horizon manner [4]. Its implementation hence hinges on the online solution of a nonlinear constrained optimization problem. A simple approach is to do model linearization and solve the resultant linear model predictive control problem by a Newton-type method, if the system is subject to only linear constraints [5], [6]. Further, nonlinear programming must be considered when generic nonlinear constraints are present. Two popular classes of methods to deal with it include the sequential quadratic programming and nonlinear interior-point methods [7], [8]. As practical applications often demand computational efficiency, significant research efforts have recently been dedicated to NMPC driven by fast real-time optimization. The continuation/GMRES method and various other proposed approaches of leveraging the structures of NMPC have proven useful in improving the computational speed [8]–[12]. Evolutionary algorithms, e.g., the particle swarm optimization method, have also received some attention in the literature [13], [14]. This is mainly due to their ability to achieve global optimization for nonconvex

[1]I. Askari and H. Fang are with the Department of Mechanical Engineering, University of Kansas, Lawrence, KS 66045, USA (e-mail: `askari`, `fang@ku.edu`).

[2]S. Zeng is with the Department of Electrical and System Engineering, Washington University, St. Louis, MO 63130, USA (e-mail: `s.zeng@wustl.edu`) and was supported in part by the NSF grant CMMI-1933976.

nonlinear programs, even though they are computationally expensive.

Despite the advancements, the high computational costs of nonlinear constrained optimization remains a bottleneck for the application of NMPC, especially when it comes to high-dimensional or highly nonlinear systems. Another concern lies in whether the global minimum can be attained, given that the optimization is nonconvex in many cases. The particle filtering approach, as a sequential Monte Carlo sampling method, has shown effective in handling the complexity of nonlinear systems in state estimation tasks [15], [16]. Its utility can be extended to tackle the above issues facing NMPC. The study in [17] illustrates particle filtering as a means to implement NMPC, by estimating the optimal control input sequence from the reference signal to track over the receding horizon. However, it is the only work on this subject to our knowledge, though a few other studies use particle filtering for the purpose of state estimation in output-feedback NMPC [18], [19]. Another related line of research is to solve NMPC using the Monte Carlo sampling-based optimization in place of numerical optimization [20], [21].

It is noteworthy that the method offered in [17] adopts an existing particle filter algorithm and thus cannot deal with generic state-input inequality constraints despite their presence in various practical systems. In this work, we aim to overcome this limitation and make the following contributions:

- We introduce a systematized formulation of NMPC through the lens of Bayesian estimation and an implementation based on vanilla particle filtering/smoothing.
- Further, we propose a novel constraint-aware particle filtering/smoothing approach using the barrier function method and then apply it to develop a new NMPC algorithm.

The proposed algorithm makes a crucial advancement of the particle-filtering-based NMPC and can find prospective use in a broad range of applications.

The rest of the paper is organized as follows. Section II interprets the NMPC problem as a Bayesian estimation problem. Section III presents a realization of NMPC based on particle filtering/smoothing and proposes the constraint-aware particle filtering/smoothing method to implement NMPC. Section IV evaluates the proposed algorithm through a simulation example. Finally, Section V concludes the paper.

II. NMPC THROUGH THE LENS OF BAYESIAN ESTIMATION

In this section, we present the NMPC problem and then examine it from the perspective of Bayesian estimation.

Consider the following discrete-time nonlinear system:

$$x_{k+1} = f(x_k, u_k), \quad (1)$$

where $x_k \in \mathbb{R}^{n_x}$ is the system state, $u_k \in \mathbb{R}^{n_u}$ is the control input, and the mapping $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ represents the state transition function. The system is subject to the following inequality constraints:

$$g_j(x, u) \leq 0, \ \forall j = 1, \ldots, m, \quad (2)$$

where m is the total number of constraints. We suppose that the control objective is to make x_k track a reference signal r_k. A corresponding NMPC problem can be stated as follows:

$$\min_{u_{k:k+H}} \ \sum_{t=k}^{k+H} \|r_t - x_t\|_Q^2 + \|u_t\|_R^2, \quad (3a)$$

$$\text{s.t.} \quad x_{t+1} = f(x_t, u_t), \quad (3b)$$

$$g_j(x_t, u_t) \leq 0 \quad \forall j = 1, \ldots, m, \quad (3c)$$

$$t = k, \ldots, k + H,$$

where H is the length of the upcoming horizon, $u_{k:k+H} = \{u_k, u_{k+1}, \ldots, u_{k+H}\}$, Q and R are weighting matrices, and x_k is known. The above problem seeks to determine the optimal control input sequence $u_{k:k+H}^*$ to minimize an objective function, which is a weighted quadratic sum of the control cost and tracking error over H steps into the future. The literature has conventionally resorted to numerical optimization approaches to compute $u_{k:k+H}^*$. Once the optimization is done, the first element of $u_{k:k+H}^*$, u_k^*, will be applied to control the system, with the rest discarded. The same optimization and control procedure will repeat itself recursively at the future time instants. To sum up, NMPC, at its core, pursues model-based predictive optimization of a system's operation in a receding horizon.

Another perspective to investigate NMPC is based on optimal estimation. The overarching notion is to interpret the NMPC problem as a problem of estimating $x_{k:k+H}$ and $u_{k:k+H}$ from $r_{k:k+H}$. Specifically, we can view $x_{k:k+H}$ and $u_{k:k+H}$ together as the state characteristic of a virtual dynamic system and $r_{k:k+H}$ as the virtual measurements made on this system. The virtual system can be expressed as

$$\begin{cases} x_{t+1} = f(x_t, u_t), \\ u_{t+1} = w_t, \\ r_t = x_t + v_t, \end{cases} \quad (4)$$

for $t = k, \ldots, k + H$, where w_t and v_t are additive disturbances. Given (4), we can consider a moving horizon estimation (MHE) problem [22] to estimate the combined state x_t and u_t:

$$\min_{x_{k:k+H}, u_{k:k+H}} \ \sum_{t=k}^{k+H} \|v_t\|_Q^2 + \|w_t\|_R^2, \quad (5a)$$

$$\text{s.t.} \quad x_{t+1} = f(x_t, u_t), \quad (5b)$$

$$u_{t+1} = w_t, \quad (5c)$$

$$r_t = x_t + v_t, \quad (5d)$$

$$g_j(x_t, u_t) \leq 0 \quad \forall j = 1, \ldots, m, \quad (5e)$$

$$t = k, \ldots, k + H.$$

The formulation in (5) is equivalent to (3), suggesting the viability of treating NMPC as an estimation problem. This view opens up a different way of dealing with NMPC.

Bayesian estimation offers another means of performing state estimation for (4). In principle, we consider the probability distribution $p(x_{k:k+H}, u_{k:k+H} \mid r_{k:k+H})$, which characterizes the information that $r_{k:k+H}$ contains about $x_{k:k+H}$ and $u_{k:k+H}$. Using Bayes' rule, we have

$$p(x_{k:k+H}, u_{k:k+H} \mid r_{k:k+H})$$
$$\propto \prod_{t=k}^{k+H} p(r_t \mid x_t) p(u_t) \cdot \prod_{t=k}^{k+H-1} p(x_{t+1} \mid x_t, u_t) \cdot p(x_k)$$
$$= \prod_{t=k}^{k+H} p(r_t \mid x_t) p(u_t), \quad (6)$$

where the last equality holds because x_k is known and $x_{t+1} = f(x_t, u_t)$ is deterministic. By (6), if assuming $p(r_t \mid x_t) \sim \mathcal{N}\left(x_t, Q^{-1}\right)$ and $p(u_t) \sim \mathcal{N}\left(0, R^{-1}\right)$ and considering the maximum likelihood estimation based on

$$\max_{x_{k:k+H}, u_{k:k+H}} \ \log p(x_{k:k+H}, u_{k:k+H} \mid r_{k:k+H}),$$

we will obtain the same formulation as in (5) except without the inequality constraints.

Bayesian estimation encompasses different ways of implementation. Among them, particle filtering is one of the most powerful approaches, which uses sequential Monte Carlo sampling to approximate the conditional probability distribution of a system's state given the measurements. The study in [17] leverages particle filtering to develop an NMPC method. However, by design, it is unable to handle the state-input inequality constraints (2), thus limiting its applicability to many practical systems. To overcome this limitation, we will propose a constraint-aware particle filtering method and exploit to enable NMPC, with details to be offered in the next sections.

Remark 1: The execution of NMPC has conventionally relied on numerical optimization. However, for high-dimensional or highly nonlinear systems, the computational complexity can be extremely high and even prohibitive, and the implementation (e.g., coding and testing) rather burdensome. By contrast, Monte Carlo sampling is computationally more efficient and easier to implement in such scenarios. Some recent studies have also pointed out that sampling can be a more favorable choice than numerical optimization for large-scale tasks [23]. Sampling-based NMPC hence holds significant promise for controlling complex systems that emerge in various problem domains.

III. NMPC VIA CONSTRAINT-AWARE PARTICLE FILTERING/SMOOTHING

In this section, we first present the problem of addressing NMPC via particle filtering/smoothing and show a basic approach to this end. Then, we develop the constraint-aware

particle filtering/smoothing and apply it to NMPC with generic inequality constraints.

A. Problem Formulation

We consider the NMPC-oriented virtual system in (4) and rewrite it compactly as

$$\begin{cases} \bar{x}_{t+1} = \bar{f}(\bar{x}_t) + \bar{w}_t, \\ r_t = M\bar{x}_t + v_t, \end{cases} \tag{7}$$

for $t = k, \dots, k + H$, where $\bar{x}_t = \begin{bmatrix} x_t^\top & u_t^\top \end{bmatrix}^\top$, $\bar{w}_t = \begin{bmatrix} 0^\top & w_t^\top \end{bmatrix}^\top$, $\bar{f}$ stems from f, and $M = [I\ 0]$. For now, we neglect the inequality constraints and will come back to this point in Section III-C. As discussed in Section II, the original NMPC problem could be cast as estimating $\bar{x}_{k:k+H}$ from $r_{k:k+H}$. According to the principle of Bayesian estimation, it is of interest to determine $p(\bar{x}_{k:k+H} \mid r_{k:k+H})$. We hence consider the following recurrence relation

$$p(\bar{x}_{k:t} \mid r_{k:t}) \propto p(r_t \mid \bar{x}_{k:t}) p(\bar{x}_t \mid \bar{x}_{t-1}) p(\bar{x}_{k:t-1} \mid r_{k:t-1}), \tag{8}$$

for $t = k, \dots, k + H$. This relation shows a forward pass from $p(\bar{x}_{k:t-1} \mid r_{k:t-1})$ to $p(\bar{x}_{k:t} \mid r_{k:t})$ and is known as *filtering*. Since NMPC only applies the first element of the computed control input sequence, we further need to consider $p(\bar{x}_k \mid r_{k:k+H})$. A backward recursion as follows is necessary to this end:

$$p(\bar{x}_t | r_{k:k+H}) = p(\bar{x}_t | r_{k:t}) \int \frac{p(\bar{x}_{t+1} | \bar{x}_t)}{p(\bar{x}_{t+1} | r_{k:t})} \times p(\bar{x}_{t+1} | r_{k:k+H}) d\bar{x}_{t+1}, \tag{9}$$

for $t = k + H, \dots, k$, which describes a backward pass from $p(\bar{x}_{t+1} \mid r_{k:k+H})$ to $p(\bar{x}_t \mid r_{k:k+H})$ and is known as *smoothing*. Together, (8)-(9) form a Bayesian forward filtering/backward smoothing framework to realize NMPC. However, a closed-form solution to $p(\bar{x}_k | r_{k:k+H})$ is generally intractable to derive for nonlinear systems. This motivates the use of sequential Monte Carlo sampling, which leads to particle filtering/smoothing.

B. NMPC via Vanilla Particle Filtering/Smoothing

To begin with, we discuss the forward filtering and consider $p(\bar{x}_{k:t} \mid r_{k:t})$ for $k \leq t \leq k + H$. Given that an analytical expression is unavailable for $p(\bar{x}_{k:t} \mid r_{k:t})$, we seek to approximate it by developing a sample-based empirical distribution. Nonetheless, it is non-trivial to draw the samples since $p(\bar{x}_{k:t} \mid r_{k:t})$ is unknown. To deal with this problem, one can leverage the technique of importance sampling. The main idea lies in drawing samples from an alternative known distribution $q(\bar{x}_{k:t} \mid r_{k:t})$, which is called importance distribution, and then evaluating the weights of the samples in relation to $p(\bar{x}_{k:t} \mid r_{k:t})$. Suppose that N samples, $\bar{x}_{k:t}^i$ for $i = 1, \dots, N$, are drawn from $q(\bar{x}_{k:t} \mid r_{k:t})$. Their importance weights are then given by

$$W_t^i = \frac{p(\bar{x}_{k:t} \mid r_{k:t})}{q(\bar{x}_{k:t} \mid r_{k:t})}. \tag{10}$$

If W_t^i for $i = 1, \dots, N$ are normalized to be between 0 and 1, then $p(\bar{x}_{k:t} \mid r_{k:t})$ is approximated as

$$p(\bar{x}_{k:t} \mid r_{k:t}) \approx \sum_{i=1}^{N} W_k^i \delta \left(\bar{x}_{k:t} - \bar{x}_{k:t}^i \right).$$

Note that (10) can be written as

$$W_t^i = \frac{p(\bar{x}_{k:t} \mid r_{k:t})}{q(\bar{x}_{k:t} \mid r_{k:t})} = \frac{p(r_t \mid \bar{x}_t^i) p(\bar{x}_t^i \mid \bar{x}_{t-1}^i)}{q(\bar{x}_t^i \mid \bar{x}_{t-1}^i, r_{k:t})} W_{t-1}^i,$$

which suggests a recursive update of W_t^i.

There are different ways to select the importance distribution q. In general, it should be chosen such that it is well related with the target distribution p and allows samples to be easily drawn. A straightforward choice is to let $q(\bar{x}_t \mid \bar{x}_{t-1}, r_{k:t}) = p(\bar{x}_t \mid \bar{x}_{t-1})$. This implies that, at time t, we can draw samples $\bar{x}_t^i \sim p(\bar{x}_t \mid \bar{x}_{t-1}^i)$, with associated normalized weights computed by

$$W_t^i = \frac{p(r_t \mid \bar{x}_t^i)}{\sum_{j=1}^{N} p(r_t \mid \bar{x}_t^j)}. \tag{11}$$

The resulting implementation is called the *bootstrap particle filter*. A common challenge in particle filtering is particle degeneracy, where a majority of the particles have zero or almost zero weights after a few time steps. The overall quality of the particles become extremely low and reduce the estimation performance. A useful method to resolve this problem is resampling, which replaces low-weight particles with those that have high weights [24].

Then, we perform the backward smoothing, which will provide a more accurate estimation of $\bar{x}_k$. Rewrite (9) as:

$$p(\bar{x}_t \mid r_{k:k+H}) = p(\bar{x}_t \mid r_{k:t}) \int \frac{p(\bar{x}_{t+1} \mid \bar{x}_t)}{\int p(\bar{x}_{t+1} \mid \bar{x}_t) p(\bar{x}_t \mid r_{k:t}) d\bar{x}_t} \times p(\bar{x}_{t+1} \mid r_{k:k+H}) d\bar{x}_{t+1}, \tag{12}$$

where the relation $p(\bar{x}_{t+1} \mid r_{k:t}) = \int p(\bar{x}_{t+1} \mid \bar{x}_t) p(\bar{x}_t \mid r_{k:t}) d\bar{x}_t$ is used. It is noted that all the probability distributions in (12) can be approximated by empirical distributions based on samples computed in the filtering procedure. Hence, according to (12), we only need to reweight the samples in smoothing:

$$W_{t|k+H}^i = \sum_{j=1}^{N} W_{t+1|k+H}^i \frac{W_t^i\, p(\bar{x}_{t+1}^j | \bar{x}_t^i)}{\sum_{l=1}^{N} W_t^l p(\bar{x}_{t+1}^j \mid \bar{x}_t^l)}, \tag{13}$$

where $W_{k+H|k+H}^i = W_{k+H}^i$. The resultant procedure is called *reweighted particle smoother*. The smoothing will end up with the following empirical distribution for $p(\bar{x}_k|r_{k:k+H})$:

$$p(\bar{x}_k \mid r_{k:k+H}) \approx \sum_{i=1}^{N} W_{k|k+H}^i \delta(\bar{x}_k - \bar{x}_k^i).$$

This implies that the best estimate of $\bar{x}_k$ from $r_{k:k+H}$ is

$$\hat{\bar{x}}_k^* = \sum_{i=1}^{N} W_{k|k+H}^i \bar{x}_k^i, \tag{14}$$
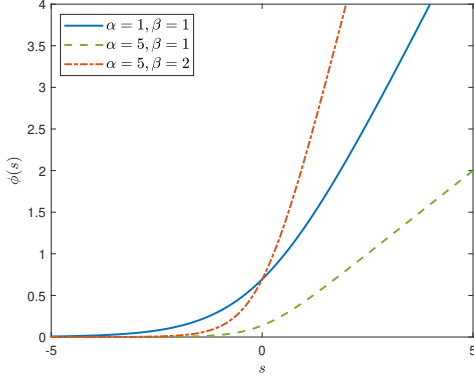
which then gives the optimal control input at time k.

Fig. 1: The softplus barrier function for different values of (α, β).

The above introduces a vanilla particle filtering/smoothing algorithm to implement NMPC. It bears a resemblance to the algorithm in [17] but uses the reweighted particle smoother to enhance the accuracy of smoothing. However, both algorithms, at their core, do not take the inequality constraints into account. Next, we propose a constraint-aware particle filtering/smoothing approach to remedy this issue.

C. Constraint-Aware Particle Filtering/Smoothing for NMPC

Here, we develop an effective method to enable particle filtering/smoothing with an awareness of the inequality constraints. The idea lies in leveraging a barrier function to create virtual measurements about the constraint satisfaction and then incorporate them into the estimation procedure.

We first construct a virtual measurement equation as

$$z_t = \phi\left(g(\bar{x}_t)\right) + \eta_t, \tag{15}$$

where z is the virtual measurement variable quantifying the constraint satisfaction, g is the collection of g_j for $j = 1, \ldots, m$, η is an additive small noise, and ϕ is a barrier function. Then, (15) will be used to evaluate the weights of the particles and penalize those failing to satisfy the constraints.

In general, a barrier function outputs almost zero at a point within a constraint set, almost infinity near the inner boundary of the set, and infinity outside the set [25]. However, such a barrier function will assign just zero weights to samples that violate the constraints. This may further deteriorate the particle degeneracy issue, impoverishing the entire ensemble of samples. Meanwhile, from the viewpoint of estimation, one only needs to make the final aggregated state estimate meet the constraints. Therefore, we depart from the traditional way to consider a fully continuous barrier function, which is designed to output almost zero within the constraint set and very large values outside the set. To this end, we choose the softplus function as a barrier function, which is expressed as

$$\phi(s) = \frac{1}{\alpha} \ln\left(1 + \exp(\beta \cdot s)\right), \tag{16}$$

where α and β are parameters to adjust the effect of the constraint violation on the particle weights. Fig. 1 shows the shape of the function under different choices of α and β. The virtual measurement z_t is accordingly set to be 0.

Now, let us add (15) to (7) and consider $p(\bar{x}_{k:t} \mid r_{k:t}, z_{k:t})$ for constraint-aware filtering. Assuming that $p(r_t, z_t \mid \bar{x}_t) = p(r_t \mid \bar{x}_t)p(z_t \mid \bar{x}_t)$, we can remold (8) as

$$p(\bar{x}_{k:t} \mid r_{k:t}, z_{k:t}) \propto p(r_t \mid \bar{x}_t)p(z_t \mid \bar{x}_t)p(\bar{x}_t \mid \bar{x}_{t-1})$$
$$\cdot p(\bar{x}_{k:t-1} \mid r_{k:t-1}, z_{k:t-1}).$$

Then, the bootstrap particle filter described in Section III-B can then be modified. Specifically, we still draw samples $\bar{x}_t^i \sim p(\bar{x}_t \mid \bar{x}_{t-1}^i)$ for $i = 1, \ldots, N$ at time t but compute the weights via

$$W_t^i = \frac{p(r_t \mid \bar{x}_t^i)p(z_t \mid \bar{x}_t^i)}{\sum_{j=1}^N p(r_t \mid \bar{x}_t^i)p(z_t \mid \bar{x}_t^j)}. \tag{17}$$

With this change, the bootstrap particle filter diminishes the weights of the samples in violation of the constraints. Once the constraint awareness has been infused into the filtering procedure, we can apply the reweighted particle smoother without change. This can be seen from the fact that (13) will not change if we consider $p(\bar{x}_t \mid r_{k:k+H}, z_{k:k+H})$ for the backward smoothing. Summarizing the above, we can readily formulate the new sampling-based NMPC algorithm based on the proposed constraint-aware particle filtering/smoothing method, which is named as CAP-NMPC outlined in Algorithm 1.

Remark 2: Particle filtering for constrained systems has received attention in several studies. However, the proposed barrier function method offers two distinct benefits. First, using the suggested barrier function, the particle filtering does not simply discard the particles violating the constraints and instead include the constraint satisfaction or violation into the weighting process. This will avoid complicating the particle degeneracy problem and balance with the need of approximating the state's posterior distribution. The treatment contrasts with the acceptance/rejection method in [26]. Second, the computational cost will see only very mild increase, even when the number of the constraints is large or the constraints are highly nonlinear, as opposed to the constrained optimization method in [27]. Finally, we highlight that the barrier function method can be easily integrated into almost every particle filter algorithm for broader application.

IV. NUMERICAL SIMULATION

In this section, we study the performance of the proposed algorithm for a path following problem of an autonomous vehicle. The code is available at: `https://github.com/KU-ISSL/PF_NMPC_ACC21`. The considered vehicle motion model is the nonlinear bicycle kinematic model, described in [28, p. 27]:

$$x_{k+1}^p = x_k^p + \Delta t \cdot \nu_k \cos(\psi_k + \beta_k), \tag{18a}$$
$$y_{k+1}^p = y_k^p + \Delta t \cdot \nu_k \sin(\psi_k + \beta_k), \tag{18b}$$
$$\nu_{k+1} = \nu_k + \Delta t \cdot a_k, \tag{18c}$$
$$\psi_k = \psi_{k-1} + \Delta t \cdot \frac{\nu_k}{l_r} \sin(\beta_k), \tag{18d}$$

Algorithm 1 CAP-NMPC: NMPC Based on Constraint-Aware Particle Filtering/Smoothing

---

1: Set up NMPC by specifying the dynamic system (1), the inequality constraints (2), the reference signal $r_{1:T}$, and the weighting matrices Q and R
2: Recast NMPC as particle filtering/smoothing by setting up the virtual system (7) and (15), and specifying $p(\bar{x}_k)$, $p(\bar{x}_{k+1} \,|\, \bar{x}_k)$, $p(r_k \,|\, \bar{x}_k)$ and $p(z_k \,|\, \bar{x}_k)$
3: **for** $k = 1, \ldots, T$ **do**
    Forward filtering
4:     **for** $t = k, \ldots, k + H$ **do**
5:       **if** $t = k$ **then**
6:         Draw samples $\bar{x}_k^i \sim p(\bar{x}_k)$, $i = 1, \ldots, N$
7:       **else**
8:         Draw samples $\bar{x}_t^i \sim p(\bar{x}_t \,|\, \bar{x}_{t-1}^i)$, $i = 1, \ldots, N$
9:         Evaluate sample weights via (17)
10:         Do resampling based on the weights
11:       **end if**
12:     **end for**
    Backward smoothing
13:     **for** $t = k + H, \ldots, k$ **do**
14:       **if** $t = k + H$ **then**
15:         Assign $W_{k+H|k+H}^i = W_{k+H}^i$, $i = 1, \ldots, N$
16:       **else**
17:         Reweight the particles using (13)
18:       **end if**
19:     **end for**
20:     Compute the optimal estimation of $\hat{\bar{x}}_k^*$ via (14)
21:   Export u_k^*, and apply it to the system (1)
22: **end for**

$$\beta_k = \tan^{-1}\left(\frac{l_r}{l_r + l_f} \tan(\delta_{f,k}) \right), \qquad (18e)$$

where x_k^p and y_k^p denote the position of the center of mass, ν_k is the vehicle's speed, ψ_k is its heading angle, and β_k is the side-slip angle. The lengths l_r and l_f represent the distance from the rear and front axles to the center of mass. The control sequence used to control the vehicle is $u_k = [a_k \; \delta_{f,k}]^\top$, where a_k and $\delta_{f,k}$ are the acceleration and front-wheel steering angle, respectively. For the purpose of comparison, we run the vanilla particle filtering/smoothing approach in Section III-B as a benchmark, which is labeled as P-NMPC for convenience. To ensure the fairness of comparison, we require that both CAP-NMPC and P-NMPC use the same set of particles with $N = 100$ throughout the simulation run. The setting used for the simulation is as follows:

- The vehicle has to follow a sinusoidal track with the x-position (x^{track}) ranging from 0 to 33 m with steps size $dx^{\text{track}} = 0.6$ m, and the y-position as $y^{\text{track}} = 2\sin(0.2x^{\text{track}})$;
- The vehicle is initialized from the state $x_0 = [-0.5, \; -0.5, \; 3, \; \pi/4]^\top$, where a deviation from the desired trajectory is introduced to better evaluate the performance of the reference tracking control;
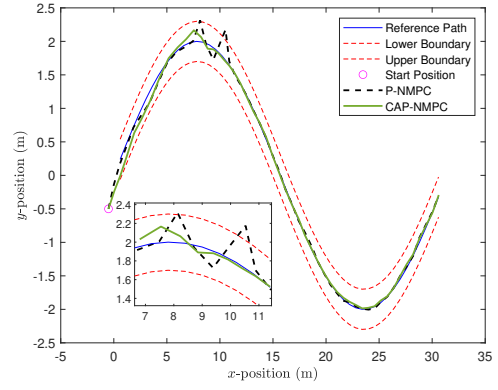
(a) Trajectories of the autonomous vehicle.
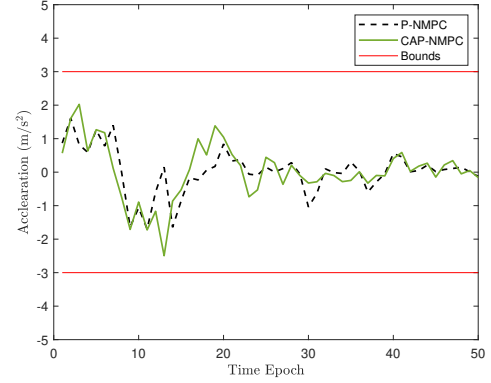
(b) Acceleration controls applied to the autonomous vehicle.
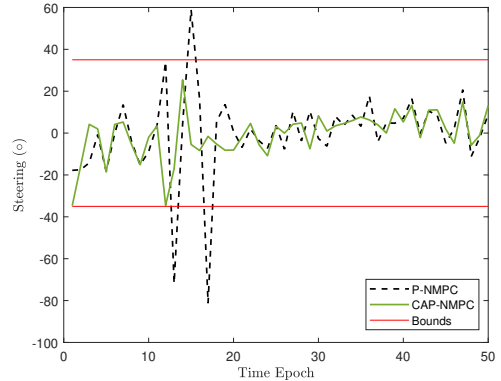
(c) Steering controls applied to the autonomous vehicle.

Fig. 2: Autonomous vehicle path following results.

- The control input is constrained between the lower and upper bounds, $\underline{u} = [-3 \text{ m/s}^2, \; -35°]^\top$ and $\overline{u} = [3 \text{ m/s}^2, \; 35°]^\top$, respectively;
- The state is constrained by the track boundaries, where the lower and upper track boundaries are each 0.3 m apart from the center of the track;
- The disturbance $\bar{w}_k$, v_k, and η_k are assumed to have Gaussian distributions with covariances as $Q_{\bar{w}} = \text{diag}(0, 0, 0, 0, 0.8, 0.4)$, $Q_v = \text{diag}(0.01, 0.01, 0, 0, 0, 0)$, and $Q_\eta = 0.01 I_5$, respectively. This setting corresponds to having $R = \text{diag}(1.25, 2.5)$ and $Q = \text{diag}(100, 100)$ for the NMPC problem in (5);

TABLE 1: RMSE and control cost achieved.

	RMSE	Cost
P-NMPC	0.330	1947
CAP-NMPC	0.324	1862

- The matrices $Q_{\bar{w}}$ and Q_v are slightly perturbed to ensure positive definiteness for numerical stability;
- The horizon length $H = 4$;
- The barrier function parameters are set as $\alpha = 5$ and $\beta = 3$;
- The axles are equidistant from the center of mass, with $l_r = l_f = 0.5$ m.

The root mean squared error (RMSE) is used to evaluate the tracking performance of the vehicle. The vehicle's trajectory is depicted in Fig. 2a with the zoomed region in the sub-axes showing an instance where the CAP-NMPC is able to satisfy the constraints while P-NMPC fails. Bearing in mind that both algorithms use the same set of particles for every horizon, it becomes evident from Fig 2 that the CAP-NMPC has successfully managed to finely tune the particle weights so that the vehicle can run while staying within the pre-set trajectory boundaries. Similarly, the control acceleration and steering inputs have also satisfied the constraints in the CAP-NMPC case, as shown in Figs. 2b-2c. The control performance is evaluated in Table 1. The CAP-NMPC outperforms the P-NMPC for both metrics. This is because it maintains an awareness of constraints. To summarize, the results show that CAP-NMPC outperforms the P-NMPC, indicating that the CAP-NMPC can sufficiently and successfully implement NMPC.

V. CONCLUSION

In this paper, we have examined NMPC from the viewpoint of Bayesian estimation and leveraged particle filtering to estimate the optimal control input of an NMPC problem. Compared with the commonly used numerical optimization, this treatment exploits Monte Carlo sampling to improve the efficiency, effectiveness, and easiness of implementing NMPC, especially conducive to the control of nonlinear systems. To endow the particle-filtering-based NMPC with the capability of handling generic state-input inequality constraints, we developed a constraint-aware particle filtering/smoothing approach based on the barrier function method. We evaluated the proposed NMPC algorithm via a simulation study of path following for autonomous vehicles. Our future work will include applying the algorithm to more complex nonlinear systems and using more sophisticated particle filtering to enhance the algorithm.

REFERENCES

[1] F. Allgower and A. Zheng, *Nonlinear Model Predictive Control*. Birkhäuser Basel, 2000.

[2] S. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Engineering Practice*, vol. 11, no. 7, pp. 733 – 764, 2003.

[3] S. V. Raković and W. S. Levine, *Handbook of Model Predictive Control*. Birkhäuser Basel, 2019.

[4] T. A. Johansen, "Introduction to nonlinear model predictive control and moving horizon estimation," in *Selected Topics on Constrained and Nonlinear Control*. STU Bratislava - NTNU Trondheim, 2011, ch. 5, pp. 187–233.

[5] W. C. Li and L. T. Biegler, "Process control strategies for constrained nonlinear systems," *Industrial & Engineering Chemistry Research*, vol. 27, no. 8, pp. 1421–1433, 1988.

[6] D. D. Brengel and W. D. Seider, "Multistep nonlinear predictive controller," *Industrial & Engineering Chemistry Research*, vol. 28, no. 12, pp. 1812–1822, 1989.

[7] M. Diehl, H. J. Ferreau, and N. Haverbeke, *Efficient Numerical Methods for Nonlinear MPC and Moving Horizon Estimation*. Springer Berlin Heidelberg, 2009, pp. 391–417.

[8] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model predictive control : theory, computation, and design*, 2nd ed. Nob Hill Publishing, 2017.

[9] T. Ohtsuka, "A continuation/GMRES method for fast computation of nonlinear receding horizon control," *Automatica*, vol. 40, no. 4, pp. 563 – 574, 2004.

[10] C. Shen, B. Buckham, and Y. Shi, "Modified C/GMRES algorithm for fast nonlinear model predictive tracking control of AUVs," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 5, pp. 1896–1904, 2017.

[11] Y. Wang and S. Boyd, "Fast model predictive control using online optimization," *IEEE Transactions on Control Systems Technology*, vol. 18, no. 2, pp. 267–278, 2010.

[12] L. T. Biegler, "Efficient solution of dynamic optimization and nmpc problems," in *Nonlinear Model Predictive Control*. Basel: Birkhäuser Basel, 2000, pp. 219–243.

[13] X. Wang and J. Xiao, "PSO-based model predictive control for nonlinear processes," in *Advances in Natural Computation*. Springer Berlin Heidelberg, 2005, pp. 196–203.

[14] X. Du, K. K. K. Htet, and K. K. Tan, "Development of a genetic-algorithm-based nonlinear model predictive control scheme on velocity and steering of autonomous vehicles," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 11, pp. 6970–6977, 2016.

[15] A. Doucet, N. De Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*. Springer New York ; London, 2001.

[16] S. Särkkä, *Bayesian Filtering and Smoothing*. Cambridge University Press, 2013.

[17] D. Stahl and J. Hauth, "PF-MPC: Particle filter-model predictive control," *Systems & Control Letters*, vol. 60, no. 8, pp. 632–643, 2011.

[18] S. K. Botchu and S. Ungarala, "Nonlinear model predictive control based on sequential monte carlo state estimation," *IFAC Proceedings Volumes*, vol. 40, no. 5, pp. 29 – 34, 2007, 8th IFAC Symposium on Dynamics and Control of Process Systems.

[19] L. Blackmore, *A Probabilistic Particle Control Approach to Optimal, Robust Predictive Control*, 2006, pp. AIAA 2006–6240.

[20] N. Kantas, J. M. Maciejowski, and A. Lecchini-Visintini, *Sequential Monte Carlo for Model Predictive Control*. Springer Berlin Heidelberg, 2009, pp. 263–273.

[21] J. M. Maciejowski, A. L. Visintini, and J. Lygeros, *NMPC for Complex Stochastic Systems Using a Markov Chain Monte Carlo Approach*. Springer Berlin Heidelberg, 2007, pp. 269–281.

[22] C. V. Rao, J. B. Rawlings, and D. Q. Mayne, "Constrained state estimation for nonlinear discrete-time systems: stability and moving horizon approximations," *IEEE Transactions on Automatic Control*, vol. 48, no. 2, pp. 246–258, 2003.

[23] Y.-A. Ma, Y. Chen, C. Jin, N. Flammarion, and M. I. Jordan, "Sampling can be faster than optimization," *Proceedings of the National Academy of Sciences*, vol. 116, no. 42, pp. 20 881–20 885, 2019.

[24] G. Kitagawa, "Monte carlo filter and smoother for non-gaussian nonlinear state space models," *Journal of Computational and Graphical Statistics*, vol. 5, no. 1, pp. 1–25, 1996.

[25] F. Borrelli, A. Bemporad, and M. Morari, *Predictive Control for Linear and Hybrid Systems*. Cambridge University Press, 2017.

[26] L. Lang, W. shiang Chen, B. R. Bakshi, P. K. Goel, and S. Ungarala, "Bayesian estimation via sequential monte carlo sampling—constrained dynamic systems," *Automatica*, vol. 43, no. 9, pp. 1615–1622, 2007.

[27] X. Shao, B. Huang, and J. M. Lee, "Constrained bayesian state estimation – a comparative study and a new particle filter based approach," *Journal of Process Control*, vol. 20, no. 2, pp. 143 – 157, 2010.

[28] R. Rajamani, *Vehicle Dynamics and Control*. Springer US, 2012.