Adaptive Horizon Multistage Nonlinear Model Predictive Control

Zawadi Mdoe, Dinesh Krishnamoorthy and Johannes Jäschke*

***Abstract*— In this paper, we present a computationally efficient multistage nonlinear model predictive controller (NMPC) with a prediction horizon update using nonlinear programming (NLP) sensitivities. Computational delay is minimized by updating the prediction horizon to a sufficient length at every time step. For a set-point tracking multistage NMPC, we first determine a terminal region around an optimal equilibrium point for each uncertainty realization in offline mode. Then using NLP-sensitivity we estimate a sufficient horizon length for the next time step such that all scenarios will be driven into their respective terminal regions. This adaptive horizon multistage NMPC (AH-*ms*NMPC) is recursively feasible and input-to-state practically stable. In a simulation study, the AH-*ms*NMPC was used to control a benchmark cooled CSTR process under parametric uncertainty. The AH-*ms*NMPC computations take 1.4% and 29.5% of the sampling interval duration for robust horizon of 1 and 2, respectively. With a robust horizon length of 1 the controller is 15 times faster than ideal-multistage NMPC with a long enough prediction horizon. The computational delay is halved with robust horizon length of 2. The performance of the two controllers was found to be similar. The improved efficiency is vital in practice for improved control performance and closed-loop stability. It is desired for real-time optimal decision making, and also under limited computing resources such as in embedded systems.**

I. INTRODUCTION

Model predictive control (MPC), has emerged as a promising dynamic optimal control strategy in process industries. It is based on mathematical modeling and online optimization of the model to find the optimal plant inputs [1]. Process systems are strongly nonlinear systems hence the growth of interest in nonlinear MPC (NMPC). The nonlinearity introduces a high degree of complexity in the optimal control problem (OCP) making it computationally demanding.

The optimization problem in an standard MPC assumes an exact model with known future disturbances and perfect prediction.

However, chemical processes can not be perfectly modelled, and the future parameters may not exactly be known ahead of time, resulting in plant-model mismatch. Uncertainty can appear as lack of truth of the identified system parameters. It can also occur as noise, either in the measurements or from the process or as a result of unknown disturbances. If the uncertainty level is pronounced, standard NMPC declines in performance and its inherent robustness can no longer guarantee recursive feasibility. As a result stochastic NMPC including robust NMPC schemes have been proposed that explicitly takes uncertainty into account.

One of the earliest robust schemes developed was a min-max MPC [10], that optimizes the worst-case scenario cost while ensuring feasibility for all possible scenarios. This open-loop min-max MPC formulation ignores the presence of future information. A feedback min-max MPC [4] was created to resolve this issue. The robust schemes are inherently large-scale problems especially for long prediction horizons. The controller designer must sacrifice computational speed for conservativeness.

Multistage NMPC (*ms*NMPC) is a robust scheme based on stochastic programming proposed in [2]. *ms*NMPC assumes discrete values of uncertain parameter realizations and presents the future evolution of uncertainty in form of a branching scenario tree along the prediction horizon. The *ms*NMPC problem increases exponentially in size with prediction horizon length. It is shown in [2] that a shortened branching horizon called robust horizon, produces as good performance because of feedback and the moving horizon nature of MPC. However, the OCP is still at least n-times larger than a standard NMPC problem if n uncertainty realizations are assumed with a minimum robust horizon of 1. The existing complexity of NMPC plus increased problem size in *ms*NMPC gives rise to significant computational cost and associated computational delay. Computational delay is undesirable for online decision making and can lead to performance degradation and instability [16]. To improve efficiency, parallel computing can be used for example by primal decomposition of the scenario-based optimal control problem as proposed in [11].

A different approach to minimize computational delay is to select the shortest possible horizon at each iteration of the *ms*NMPC. A variable horizon method that was proposed in [4] requires solving a mixed-integer nonlinear problem for the *ms*NMPC which is even more computationally demanding than the original problem. The adaptive horizon method in [3] updates the prediction horizon length efficiently at every time step by solving an NLP-sensitivity problem online that determines the sufficient horizon length for the system to enter a pre-determined terminal region of attraction. If the control problem is such that it is attracted to a terminal region set, the new horizon will be reduced at each subsequent stage making the problem smaller and simpler to solve.

The main contribution of this paper is to extend the adaptive horizon method [3] on to multistage NMPC. We compare the performance with a standard approach and demonstrate its potential for improving computational efficiency. The rest

This work was supported by the Faculaty of Natural Sciences (NV) at the Norwegian University of Science of Technology (NTNU).

All authors are with the Department of Chemical Engineering, Norwegian University of Science and Technology, N-7491 Trondheim, Norway. (E-mail addresses: zawadi.n.mdoe@ntnu.no; dinesh.krishnamoorthy@ntnu.no; johannes.jaschke@ntnu.no)

*Johannes Jäschke is the corresponding author.

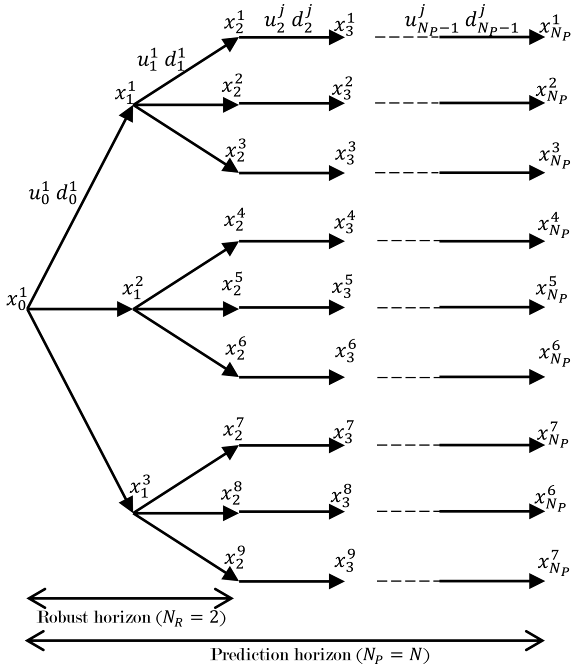

Fig. 1. A scenario tree representation of uncertainty evolution. A scenario is a path from the root node to a leaf node. There are a total of $N_S = 9$ scenarios, with a robust horizon $N_R = 2$ and prediction horizon $N_P = N$.

of this paper is organized as follows. Section II carries the main contribution of this manuscript. It sets foundation on the nonlinear problem and ideal multistage NMPC framework. We also present the adaptive horizon algorithm and the computation of terminal region of attraction. Section III presents recursive feasibility and stability proofs of the method. Section IV shows results of a simulation study on a benchmark example under parametric uncertainty. Finally, Section V presents the drawn conclusions.

II. ADAPTIVE HORIZON MULTISTAGE NMPC

We represent the dynamics of a system as a discrete-time nonlinear model given by (1).

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{d}_k) \tag{1}$$

where, $\mathbf{x}_k \in \mathcal{X} \subseteq \mathbb{R}^{n_x}$, $\mathbf{u}_k \in \mathcal{U} \subseteq \mathbb{R}^{n_u}$ are the state and input vectors, and $\mathbf{d}_k \in \mathcal{D} \subset \mathbb{R}^{n_d}$ is the vector of uncertain parameters in the plant model at time step t_k, and $k \geq 0$ is the time index.

A. Ideal Multistage NMPC

Multistage NMPC [2] employs a scenario tree to define the possible instances that an uncertain parameter can take in the future along the prediction horizon. Fig. 1 shows a scenario tree structure with a robust horizon $N_R = 2$ and a prediction horizon $N_P = N$. The root node of the tree is the current state of the plant from which the disturbance $\mathbf{d}_k$ assumes any value in the uncertainty set, $\mathcal{D}$. For instance, in Fig. 1 we may assume $\mathbf{d}_k$ can take 3 representative discrete values $\mathcal{D} = \{\text{low}, \text{nom}, \text{high}\}$. If $n_d = 1$, three different states will be predicted at the next time step, t_{k+1}. The tree will continue branching from each node, and branching stops at the robust horizon. After that, the uncertainty realization is assumed to remain constant. Then we obtain a total number of scenarios, N_S such that $N_S = |\mathcal{D}|^{N_R}$, where $|\mathcal{D}|$ is the cardinality of $\mathcal{D}$. The robust horizon is an admissible simplification due to the moving horizon aspect of the NMPC controller, see [2]. The resulting OCP (2) is re-optimized at each iteration and the optimal first step input is injected to the plant.

$$V_N^{ms}(\hat{\boldsymbol{x}}_t) = \min_{\mathbf{x}_k^j, \mathbf{u}_k^j} \quad \sum_{j \in \mathcal{S}} \Big\{ \omega^j \Big(V_f(\mathbf{x}_N^j) + \sum_{k=0}^{N-1} \ell(\mathbf{x}_k^j, \mathbf{u}_k^j) \Big) \Big\} \tag{2a}$$

$$\text{s.t.} \qquad \mathbf{x}_{k+1}^j = \mathbf{f}(\mathbf{x}_k^j, \mathbf{u}_k^j, \mathbf{d}_k^j),\ k = 0, \dots, N-1 \tag{2b}$$

$$\mathbf{x}_0^j = \mathbf{x}_0 = \hat{\boldsymbol{x}}_t \tag{2c}$$

$$\mathbf{u}_k^j = \mathbf{u}_k^l \quad \text{if} \quad \mathbf{x}_k^j = \mathbf{x}_k^l \tag{2d}$$

$$\mathbf{x}_k^j \in \mathcal{X}, \quad \mathbf{u}_k^j \in \mathcal{U}, \quad \mathbf{x}_N^j \in \mathcal{X}_f^j \tag{2e}$$

$$j, l \in \mathcal{S} \quad \text{where} \quad \mathcal{S} = \{1, \dots, N_S\} \tag{2f}$$

where, $\hat{\boldsymbol{x}}_t$ is the state measurement of the plant at time t. $\mathbf{x}_k^j$ and $\mathbf{u}_k^j$ are the predicted state and inputs at stage k and scenario j, and $\mathbf{d}_k^j$ is the uncertain parameter of scenario j at stage k. ω_j denotes the assigned weight for scenario j. $\ell(\cdot\,,\cdot)$ is the scalar stage cost function such that $\ell : \mathcal{X} \times \mathcal{U} \rightarrow \mathbb{R}$. The weighted sum of the scenario summation of the stage costs $\ell(\cdot\,,\cdot)$ and terminal cost $V_f(.)$ is the objective value function V_N^{ms} for the multistage OCP given by (2a). $\mathcal{S}$ is an index set for the scenarios. (2b) is the system model and (2c) is the initial condition which equals the current plant state $\hat{\boldsymbol{x}}_t$. (2d) are the *non-anticipativity constraints* (NACs).

B. Adaptive horizon using NLP sensitivity

The adaptive horizon method updates the prediction horizon length of the next *ms*NMPC iteration by estimating a sufficient horizon length from the current solution using NLP-sensitivities, see [3], [9], [14].

We assume that the system is controlled with a set-point tracking objective and has an optimal equilibrium pair $(\mathbf{x}_f^j, \mathbf{u}_f^j)$ for all scenarios $j \in \mathcal{S}$. This point is an equilibrium pair such that $\mathbf{x}_f^j = \mathbf{f}(\mathbf{x}_f^j, \mathbf{u}_f^j, \mathbf{d}_f^j)$ for all $j \in \mathcal{S}$ and gives the lowest stage cost among all the equilibrium pairs, see [17].

Assumption 1: There exists a terminal region of attraction $\mathcal{X}_f^j$ around an optimal equillibrium pair $(\mathbf{x}_f^j, \mathbf{u}_f^j)$ for all $j \in \mathcal{S}$ such that

$$\mathcal{X}_f^j := \{\mathbf{x}^j \mid \|\Delta\mathbf{x}^j\| \leq c_f^j\} \tag{3}$$

where $\Delta\mathbf{x}^j = \mathbf{x}^j - \mathbf{x}_f^j$ and c_f^j is the terminal region's radius.

Assumption 2: There exists an open ball $\mathcal{B}_\epsilon(\mathbf{x}_f^j, \mathbf{u}_f^j)$ with radius $\epsilon > 0$ around the optimal equilibrium pair $(\mathbf{x}_f^j, \mathbf{u}_f^j)$, such that the stage cost is small compared to the stage cost at the initial condition for all $(\mathbf{x}^j, \mathbf{u}^j) \in \mathcal{B}_\epsilon(\mathbf{x}_f^j, \mathbf{u}_f^j)$ and for all $j \in \mathcal{S}$ [3].

Following Assumption 1, Fig. 2 illustrates possible open loop optimal trajectories for a system with 3 uncertainty

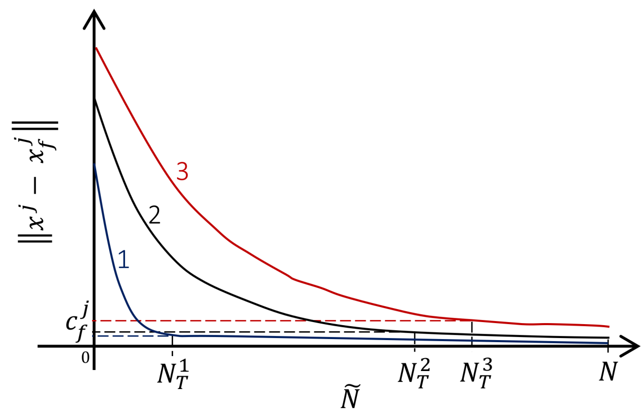

Fig. 2. Open loop trajectories for different scenarios as they approach their respective terminal regions. Their sufficient horizon lengths N_T^j are also indicated. Here the prediction horizon can be truncated at N_T^3.

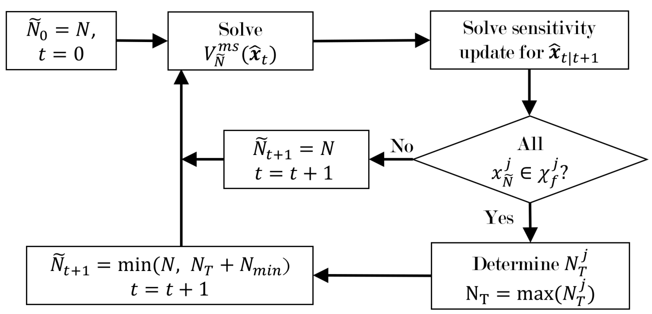

Fig. 3. Adaptive horizon algorithm to determine the sufficient horizon length for the next stage problem.

scenarios. If the prediction horizon is long enough, each scenario's optimal trajectory enters their respective terminal region at their respective stage N_T^j. There exists a prediction horizon length N_T where all scenarios would be steered into their respective terminal regions. According to Assumption 2, the prediction horizon of the OCP can be truncated at that stage. For instance, in Fig. 2 we would truncate at $N_T \geq N_T^3$.

The proposed algorithm is summarized by a flowchart in Fig. 3. The algorithm begins with choosing a safety factor $N_{\min}$, and a sufficiently long horizon N as the initial prediction horizon for (2). At time t the NLP is solved with a prediction horizon $\tilde{N}_t$. An NLP-sensitivity problem is solved to make a one-step ahead prediction of the optimal trajectory for the NLP at time $t+1$, see [3], [9], [13], [14]. This trajectory is checked if it enters the terminal region for all scenarios. If true, the sufficient horizon lengths for each scenario N_T^j are determined. The overall sufficient horizon length $N_T = \max(N_T^1, \ldots, N_T^j)$ for all $j \in \mathcal{S}$. The prediction horizon length for the next iteration is $\tilde{N}_{t+1} = N_T + N_{\min}$.

C. Quasi-infinite horizon NMPC

The adaptive horizon method requires identification of the terminal regions with radii c_f^j and a terminal cost function $V_f(\cdot)$ of the closed loop system for each scenario. This is approximated by a quasi-infinite horizon NMPC presented by [3]. In order to determine terminal regions $\mathcal{X}_f^j$ consider Assumption 3.

Assumption 3: There exists a stabilizing LQR controller with a local control law: $\Delta\mathbf{u}^j = -K^j\Delta\mathbf{x}^j$ such that $\mathbf{f}(\mathbf{x}^j, \mathbf{u}_f^j - K^j\Delta\mathbf{x}^j, \mathbf{d}^j) \in \mathcal{X}_f^j$ for all $\mathbf{x}^j \in \mathcal{X}_f^j$.

We determine a ball of radius c_f^j in the vicinity of $\mathbf{x}_f^j$ where the LQR control law stabilizes the uncertain nonlinear system (1). The terminal state and input is an optimal equilibrium pair $(\mathbf{x}_f^j, \mathbf{u}_f^j)$ obtained by solving a steady-state optimization problem at $\mathbf{d}^j$. The nonlinear system (1) is linearized about $(\mathbf{x}_f^j, \mathbf{u}_f^j)$ as follows

$$\Delta\mathbf{x}_{k+1}^j = A^j\Delta\mathbf{x}_k^j + B^j\Delta\mathbf{u}_k^j + \phi^j(\Delta\mathbf{x}_k^j, \Delta\mathbf{u}_k^j, \mathbf{d}^j) \quad (4)$$

for all $j \in \mathcal{S}$, where A^j and B^j are the jacobians of $\mathbf{f}$ w.r.t $\mathbf{x}$ and $\mathbf{u}$ respectively at $(\mathbf{x}_f^j, \mathbf{u}_f^j)$ and $\phi^j(\Delta\mathbf{x}_k^j, \Delta\mathbf{u}_k^j, \mathbf{d}^j)$ is the linearization error. An infinite horizon LQR is proposed as the stabilizing controller in the terminal region given by

$$\Delta\mathbf{x}^{j\top} P^j \Delta\mathbf{x}^j = \min \sum_{k=0}^{\infty} (\Delta\mathbf{x}_k^{j\top} Q \Delta\mathbf{x}_k^j + \Delta\mathbf{u}_k^{j\top} R \Delta\mathbf{u}_k^j) \quad (5a)$$

$$\text{s.t.} \quad \Delta\mathbf{x}_{k+1}^j = A^j\Delta\mathbf{x}_k^j + B^j\Delta\mathbf{u}_k^j \quad \forall\, k = 0 \ldots \infty \quad (5b)$$

for all $j \in \mathcal{S}$, where $P^j > 0$ is a solution of the discrete algebraic Ricatti equation.

The closed loop trajectory of the nonlinear system (1) controlled by the LQR is given by

$$\Delta\mathbf{x}_{k+1}^j = A_{cl}^j\Delta\mathbf{x}_k^j + \bar{\phi}^j(\Delta\mathbf{x}_k^j, \mathbf{d}^j) \quad (6)$$

for all $j \in \mathcal{S}$, where $A_{cl}^j = A^j - B^jK^j$ and $\bar{\phi}^j(\Delta\mathbf{x}_k^j, \mathbf{d}^j) = \phi^j(\Delta\mathbf{x}_k^j, -K^j\Delta\mathbf{x}_k^j, \mathbf{d}^j)$. The LQR controller design assumes $\phi^j(\Delta\mathbf{x}_k^j, \Delta\mathbf{u}_k^j, \mathbf{d}^j) = 0$. For the nonlinear system, we determine the bounds on the linearization error for each scenario j using Lemma 1 as follows.

Lemma 1: There exists M^j, $q \in \mathbb{R}_+$ for all $j \in \mathcal{S}$ such that

$$\|\bar{\phi}^j(\Delta\mathbf{x}_k^j, \mathbf{d}^j)\| \leq M^j\|\Delta\mathbf{x}_k^j\|^q, \quad \forall\, \mathbf{x}^j \in \mathcal{X} \quad (7)$$

Proof: See [3].

Analytical determination of the bound (7) is tedious and thus one can run one-step simulations offline. This method was proposed by [3], and is also adopted in this work. In offline mode, a random initial point in the system domain $\mathcal{X} \times \mathcal{U}$ is taken and a one-step simulation determines the linearization error by evaluating the difference (8).

$$\bar{\phi}^j(\Delta\mathbf{x}_k^j, \mathbf{d}^j) = \mathbf{f}(\mathbf{x}_k^j, \mathbf{u}_f^j - K^j\Delta\mathbf{x}_k^j, \mathbf{d}^j) - A_{cl}^j\Delta\mathbf{x}_k^j \quad (8)$$

After determining the bounds of the linearization error for each scenario j using Lemma 1, we quantify the size of the terminal region $\mathcal{X}_f^j$ using Lemma 2.

Lemma 2: Suppose Assumption 1 holds, the radius of the terminal region of attraction c_f^j for all $\mathbf{d}^j$ and for all $j \in \mathcal{S}$

is dependent on the linearization error bound and is given by

$$c_f^j := \left(\frac{-\bar{\sigma}^j \Lambda^j + \sqrt{(\bar{\sigma}^j \Lambda^j)^2 + (\underline{\lambda}_{W^j} - \epsilon_{LQ})\Lambda^j)}}{\Lambda^j M^j} \right)^{\frac{1}{q-1}} \tag{9}$$

where $\bar{\sigma}^j$ is the maximum singular value of A_{cl}^j, $\bar{\lambda}_{W^j}$ and $\underline{\lambda}_{W^j}$ are the maximum and minimum eigen values of $W^j := Q + K^{j\top} R K^j$, $\Lambda^j := \frac{\bar{\lambda}_{W^j}}{(1-\bar{\sigma}^j)^2}$, and $\epsilon_{LQ} > 0$ is a small constant: an allowable tolerance for the terminal cost V_f^j.

Proof: See [3].

III. PROPERTIES OF ADAPTIVE HORIZON MULTISTAGE NMPC

In this section, we assume a transformed problem such that the optimal equilibrium pairs $(\mathbf{x}_f^j, \mathbf{u}_f^j)$ for all $j \in \mathcal{S}$ are all centered at $(0,0)$ using $(\mathbf{x}^j, \mathbf{u}^j) = (\tilde{\mathbf{x}}^j - \tilde{\mathbf{x}}_f^j, \tilde{\mathbf{u}}^j - \tilde{\mathbf{u}}_f^j)$ where $\tilde{\mathbf{x}}^j$ and $\tilde{\mathbf{u}}^j$ are states and inputs before transformation.

A. Recursive feasibility

Assumption 4: The adaptive horizon multistage NMPC (AH-*ms*NMPC) has the following properties.

- Lipschitz continuity: The functions $\mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{d})$, $\ell(\mathbf{x}, \mathbf{u})$, and $V_f^j(\mathbf{x})$ are Lipschitz continuous with $\mathbf{f}(0, 0, \mathbf{d}^j) = 0$ for all $j \in \mathcal{S}$, $\ell(0, 0) = 0$ and $V_f^j(0) = 0$.
- Constraint set: The sets $\mathcal{X}$ and $\mathcal{X}_f \subseteq \mathcal{X}$ are closed. $\mathcal{U}$ is a compact set. All constraint sets contain the origin.
- Common terminal region: There exists a common terminal region $\mathcal{X}_f = \bigcap_{j \in \mathcal{S}} \mathcal{X}_f^j$ that is control invariant.

Theorem 1: Suppose that Assumption 4 holds, then the AH-*ms*NMPC is recursively feasible.

Proof: In [8], [12], [13] recursive feasibility of an *ms*NMPC with a constant prediction horizon was proven. If the new horizon length $\tilde{N}_{t+1}$ is sufficient to steer the system into the control-invariant common terminal region set $\mathcal{X}_f$, then recursive feasibility is proven using same arguments as standard NMPC with terminal region in [1].

B. Stability

In order to prove stability we define the following.

Definition 1: Comparison functions: A function $\alpha : \mathbb{R}_0^+ \to \mathbb{R}_0^+$ is a $\mathcal{K}$-function if it is continuous and strictly increasing with $\alpha(0) = 0$. A function α is a $\mathcal{K}_\infty$-function if $\alpha \in \mathcal{K}$ and is unbounded. A function $\beta : \mathbb{R} \times \mathbb{Z}_+ \to \mathbb{R}$ is a $\mathcal{KL}$-function if it is continuous, $\beta(\,\cdot\,, k) \in \mathcal{K}$ for all $k > 0$, and $\beta(n, \cdot\,)$ is decreasing for all $s > 0$ and $\beta(s, k) \to 0$ as $k \to \infty$.

Definition 2: Robustly Positive Invariant (RPI) set: A set $\mathcal{X}$ is an RPI set for (1) if $\mathbf{x}_{t+1} \in \mathcal{X}$ holds for all $\mathbf{x}_t \in \mathcal{X}$ and $\mathbf{d} \in \mathcal{D}$.

Definition 3: Input-to-State Practical Stability (ISpS): The system (1) is ISpS in $\mathcal{X}$ if there exists a $\mathcal{KL}$-function β, a $\mathcal{K}$-function γ, $t > 0$ and $c \geq 0$ for all $\mathbf{d} \in \mathcal{D}$ such that

$$\|\mathbf{x}_t\| \leq \beta(\|\mathbf{x}_0\|, t) + \gamma(\|\mathbf{d}\|) + c, \quad \forall \mathbf{x}_0 \in \mathcal{X} \tag{10}$$

Definition 4: ISpS-Lyapunov function: A function $V(\,\cdot\,)$ is an ISpS-Lyapunov function for the system (1) if there exists an RPI set $\mathcal{X}$, $\mathcal{K}_\infty$-functions α_1, α_2, α_3 and $\mathcal{K}$-function σ, $c_0, c_1 \geq 0$ for all $\mathbf{x} \in \mathcal{X}$ and $\mathbf{d} \in \mathcal{D}$ such that

$$\alpha_1(\|\mathbf{x}\|) \leq V(\mathbf{x}) \leq \alpha_2(\|\mathbf{x}\|) + c_0 \tag{11a}$$

$$\Delta V(\mathbf{x}, \mathbf{d}) \leq -\alpha_3(\|\mathbf{x}\|) + \sigma(\|\mathbf{d}\|) + c_1 \tag{11b}$$

Assumption 5: ISpS assumptions for AH-*ms*NMPC.

- If Assumption 3 holds then $V_f(\mathbf{f}(\mathbf{x}^j, -K^j\mathbf{x}^j, \mathbf{d}^j)) - V_f(\mathbf{x}^j) \leq -\ell(\mathbf{x}^j, -K^j\mathbf{x}^j)$ for all $\mathbf{x}^j \in \mathcal{X}_f$.
- The stage cost satisfies the following for all $\mathbf{x} \in \mathcal{X}$ and $\mathbf{d} \in \mathcal{D}$:

$$\alpha_L(\|\mathbf{x}\|) \leq \ell(\mathbf{x}, \mathbf{u}) \leq \alpha_U(\|\mathbf{x}\|) + \sigma_U(\|\mathbf{d}\|) \tag{12}$$

  where α_L, α_U and σ_U are $\mathcal{K}_\infty$-functions [13], [15].
- The solution for the AH-*ms*NMPC satisfies the Mangasarian-Fromovitz Constraint Qualification (MFCQ) and the Strong Second-Order Sufficient Conditions (SSOSC) such that the NLP-sensitivity theory [14] can be applied.

Let us define a set of acceptable horizon lengths $\mathcal{N} = \{\tilde{N} \,|\, N_{\min} \leq \tilde{N} \leq N,\ \tilde{N} \in \mathbb{Z}_+\}$ and the subset $\mathcal{N}_t \subset \mathcal{N}$ of acceptable horizon lengths for all the feasible problems of (2). We define a mapping $H : \mathbb{R}^n \times \mathcal{N} \times \mathbb{R}^n \to \mathcal{N}$ for the algorithm illustrated by Fig. 3 such that $\tilde{N}_{t+1} = H(\hat{\boldsymbol{x}}_t, \tilde{N}_t, \hat{\boldsymbol{x}}_{t+1|t}) \in \mathcal{N}_{t+1}$.

Assumption 6: If problem (2) at time t with $\hat{\boldsymbol{x}}_t$ is feasible then so is the problem at time $t+1$ with $\hat{\boldsymbol{x}}_{t+1}$ and $\tilde{N}_{t+1} = H(\hat{\boldsymbol{x}}_t, \tilde{N}_t, \hat{\boldsymbol{x}}_{t+1|t}) \in \mathcal{N}_{t+1}$ for all $\hat{\boldsymbol{x}}_t$, $\hat{\boldsymbol{x}}_{t+1|t} \in \mathcal{X}$, $\tilde{N}_t \in \mathcal{N}_t$.

Theorem 2: If $\mathcal{X}$ is an RPI set and Assumptions 4, 5, 6 hold and α_1, $\alpha_2, \alpha_3 \in \mathcal{K}_\infty$, $\sigma \in \mathcal{K}$ and $V(\,\cdot\,)$ is an ISpS-Lyapunov function with $\Delta V(\hat{\boldsymbol{x}}_t, \mathbf{d}) = V_{\tilde{N}_{t+1}}(\hat{\boldsymbol{x}}_{t+1}) - V_{\tilde{N}_t}(\hat{\boldsymbol{x}}_t)$ for all $\hat{\boldsymbol{x}}_t \in \mathcal{X}$ and $N_t \in \mathcal{N}_t$, then the AH-*ms*NMPC system is ISpS stable in $\mathcal{X}$.

Proof: For the sake of brevity, we outline a sketch of the proof. Following Assumptions 4 and 5, it is proven by [8], [13], [18] that the cost $V_N^{ms}(x)$ is an ISpS-Lyapunov function and that an ideal-*ms*NMPC results into an ISpS stable sytem. To ensure stability, [13] highlights that a terminal cost for each scenario is required, and they should be control Lyapunov functions. In the case of AH-*ms*NMPC, we have two possible cases of prediction horizon update which include:

- prediction horizon shortening i.e. $N_{t+1} < N_t$ or
- constant prediction horizon or lengthening i.e. $N_{t+1} \geq N_t$.

Using Assumptions 5 and 6 in each case above, we are always guaranteed feasibility. We predetermined the terminal regions at which we have a control Lyapunov terminal cost function. Since prediction horizon shortening occurs at the point when all scenarios have entered their respective terminal regions plus a safety factor, then the weighted terminal cost is an ISpS Lyapunov-function. This implies that AH-*ms*NMPC also results in an ISpS stable system.

IV. SIMULATION EXAMPLE

We consider a continuously-stirred tank reactor (CSTR) system with a cooling jacket from [5]. The reactions in the

TABLE I

SYSTEM PARAMETERS

Parameter	Value	Units	Parameter	Value	Units
$A_{1,2}$	9.043×10^{12}	/h	ρ	0.9342	kg/m^3
A_3	9.043×10^9	/h	A_R	0.215	m^2
$E_{1,2}/R$	9758.3	K	V_R	10.01	m^3
E_3/R	8560.0	K	T_{in}	130.0	°C
ΔH_{AB}	4.2	kJ/mol	k_W	4032	kJ/hm^{2}K
ΔH_{BC}	-11.0	kJ/mol	m_J	5	kg
ΔH_{AD}	-41.85	kJ/mol	R	8.314×10^{-3}	kJ/Kmol
c_p	3.01	kJ/kgK	$c_{p,J}$	2.0	kJ/kgK

CSTR are given by (13), where component B is the main product.

$$A \xrightarrow{k_1} B \xrightarrow{k_2} C \tag{13a}$$

$$A \xrightarrow{k_3} D \tag{13b}$$

The system dynamics of the cooled CSTR are described by the following set of equations.

$$\dot{c_A} = F(c_{A,0} - c_A) - k_1 c_A - k_3 c_A^2 \tag{14a}$$

$$\dot{c_B} = -F c_B + k_1 c_A - k_2 c_B \tag{14b}$$

$$\dot{T_R} = F(T_{\text{in}} - T_R) + \frac{k_W A_R}{\rho c_p V_R}(T_J - T_R) - \frac{k_1 c_A \Delta H_{AB} + k_2 c_B \Delta H_{BC} + k_3 c_A^2 \Delta H_{AD}}{\rho c_p} \tag{14c}$$

$$\dot{T_J} = \frac{1}{m_J c_{p,J}}(\dot{Q}_J + k_W A_R (T_R - T_J)) \tag{14d}$$

where reaction rates k_i follow Arrhenius law, $k_i = A_i \exp\left(\frac{-E_{A,i}}{RT_R}\right)$. Table I presents all the model parameters, their corresponding values and units. The states vector $\mathbf{x} = [c_A, c_B, T_R, T_J]^\top$ consists of the concentration of A and B, reactor and coolant temperature respectively. The control inputs $\mathbf{u}$, are inlet flow per reactor volume $F = V_{in}/V_R$, and cooling rate $\dot{Q}_J$.

The control objective is to maintain the concentration of B to a desired set-point. The set-point is $c_B^{set} = 0.5\,\text{mol}/\ell$ throughout the operation period. The activation energy, $E_{A,3}$ is the uncertain parameter of the system. The stage cost is a typical set-point tracking error for c_B with control movement penalization terms $\Delta F_k = F_k - F_{k-1}$ and $\Delta \dot{Q}_{J k} = \dot{Q}_{J k} - \dot{Q}_{J k-1}$ given by

$$\ell_k = (c_{B k} - c_B^{\text{set}})^2 + r_1 \Delta F_k^2 + r_2 \Delta \dot{Q}_{J k}^2 \tag{15}$$

where the control penalties $r_1 = 10^{-5}$ and $r_2 = 10^{-7}$.

Regularization terms were added to the stage cost to ensure dissipativity and strong convexity as done in [9]. The initial prediction horizon $N = 40$ and for each scenario terminal constraints are included such that $\|\mathbf{x}_N^j - \mathbf{x}_f^j\|^2 \leq {c_f^j}^2$. The robust horizon is chosen to be either 1 or 2. For $N_R = 2$ and $n_d = 1$, we obtain a 9-scenario tree structure as in Fig. 1 for the *ms*NMPC.

A. Terminal region calculations

With a selected LQR controller tuning of $Q = I_4$ and $R = [10^{-3}\, 10^{-4}]I_2$ we design a stabilizing controller for each of the 9 scenarios. As outlined before, 10^5 one-step simulations from random initial values are performed offline for each scenario. Fig. 4 is a plot of the linearization error against $\Delta\mathbf{x}$ for each scenario. The linearization errors for all the scenarios are not significantly different and therefore a common linearization error bound (black solid line) $M^j = 125$ and $q = 2.2$, is fitted. Using (9) applied to each scenario, the radius of the terminal regions were computed as c_f^1, c_f^4, $c_f^7 = 0.023$, c_f^2, c_f^5, $c_f^8 = 0.023$ and c_f^3, c_f^6, $c_f^9 = 0.024$.

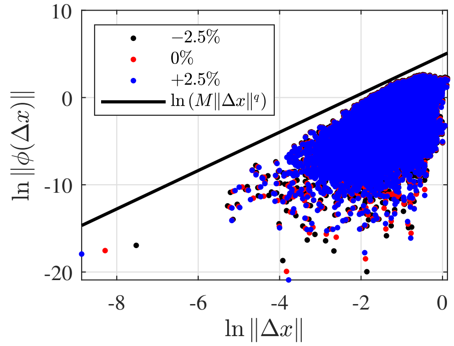

Fig. 4. Linearization bound after 10000 simulations for each parameter uncertainty realization ($\mathcal{D} = \{E_{A,3}^{\text{nom}} \pm 2.5\%\}$, $M^j = 125$ and $q = 2.2$).

B. Simulation results

The sampling time was chosen to be $0.005\,\text{h}$ and the closed loop system for (14) was simulated for an operation time of $0.3\,\text{h}$. The inlet concentration of component A is constant $c_{A,0} = 5.1\,\text{mol}/\ell$ and the uncertain parameter was $\hat{d} = E_{A,3}^{\text{nom}} + 1\%$ throughout. The control performance of the AH-*ms*NMPC was compared to an ideal-*ms*NMPC. $N_{\text{min}} = 3$ was selected as the minimum horizon length and the safety factor.

We implemented the controller using `CasADi v.3.4.5`[6] in a `MATLAB` environment. The NLP solver was `IPOPT 3.12`[7] and linear solver was `MA27` running on a 2.6 GHz Intel Core-i7 with 16GB RAM.

Fig. 5 shows the optimal state trajectories and control actions for both schemes. It can be seen that the optimal trajectories for ideal-*ms*NMPC and AH-*ms*NMPC are the same. Both controllers are capable of minimizing offset even with a plant-model mismatch in the uncertain parameter $E_{A,3}$. It is evident that the adaptive horizon scheme does not affect the control performance of the *ms*NMPC controller.

Fig. 6 illustrates how the prediction horizon length, N_t is continuously reduced to a minimum value as the system is driven to the terminal region. It also shows the amount of time either controller spent to solve the NLP at each iteration. While the ideal-*ms*NMPC continues solving the problem for an entire maximum horizon length at each iteration spending significantly larger time, the adaptive horizon method always solved a sufficiently reduced problem, making it faster.

As shown in Table II, the adaptive horizon scheme computes 15 times faster for $N_R = 1$, and computes 2 times faster for $N_R = 2$ on average per iteration compared to

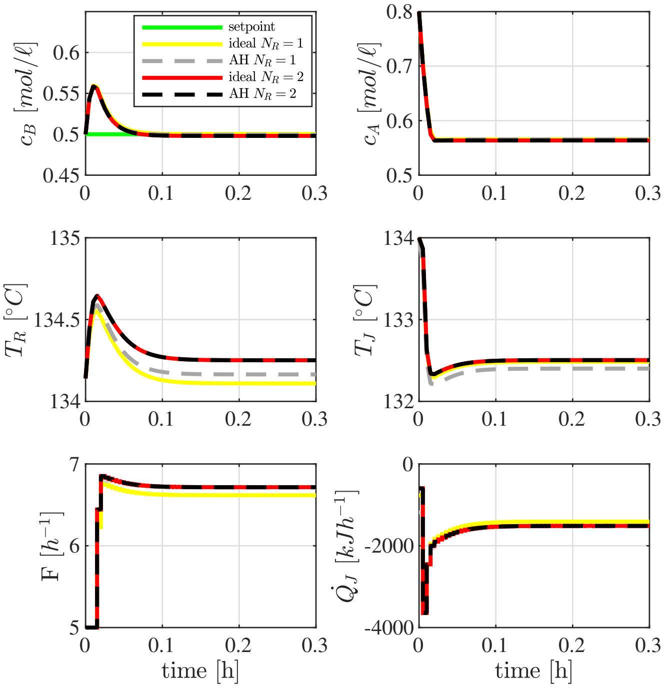

Fig. 5. Comparison between the ideal *ms*NMPC with AH*ms*NMPC with $\hat{d} = E_{A,3}^{\mathrm{nom}} + 2\%$ for both $N_R = 1$ and $N_R = 2$.

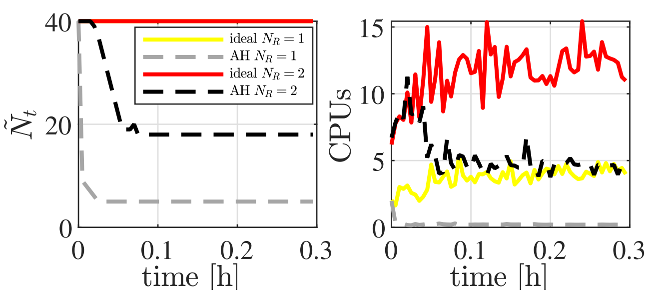

Fig. 6. Comparison of the prediction horizon length and computation time between ideal *ms*NMPC and AH*ms*NMPC with parametric uncertainty realization $\hat{d} = E_{A,3}^{\mathrm{nom}} + 1\%$ for both $N_R = 1$ and $N_R = 2$.

the ideal-*ms*NMPC. For the sampling time of 18s, the AH-msNMPC's computational delay with $N_R = 1$ is 1.4% of the sampling interval while with $N_R = 2$ is 29.5% of the sampling interval which is a significant reduction compared to the ideal-*ms*NMPC's 64% fraction.

V. CONCLUSIONS

We have presented in this paper an adaptive horizon multistage NMPC formulation that is both efficient and robust. The resulting optimal control problem is shown to be recursively feasible and input-to-state practical (ISpS) stable. A simulation case study on a CSTR shows that the control performance of the adaptive horizon multistage NMPC framework is the same as that of ideal multistage NMPC. The adaptive horizon framework is faster than the ideal-multistage NMPC. The adaptive horizon update is beneficial for multistage NMPC in order to reduce NLP size and the computational delay. This method can be applied to control a nonlinear system provided that it is attracted to a terminal region.

TABLE II

MEAN COMPUTATION TIME

N_R	Controller	CPU time [s]		
		Max	Mean	Min
1	ideal-*ms*	5.3215	3.7385	1.6401
	AH-*ms*	1.9783	0.2453	0.1500
2	ideal-*ms*	15.4335	11.5805	6.1988
	AH-*ms*	11.2770	5.3149	3.7616

ACKNOWLEDGMENT

The authors are grateful for the support of Faculty of Natural Sciences (NV) at Norwegian University of Science and Technology. We also express our gratitude towards Professor L.T. Biegler from Carnegie Mellon University for his insightful discussions.

REFERENCES

[1] J.B. Rawlings and D.Q. Mayne *Model predictive control: Theory and design*. Nob Hill Pub., 2009.
[2] S. Lucia, T. Finkler and S. Engell. "Multi-stage nonlinear model predictive control applied to a semi-batch polymerization reactor under uncertainty." *Journal of Process Control* 23.9 (2013): 1306-1319.
[3] D.W. Griffith, L.T. Biegler and S.C. Patwardhan. "Robustly stable adaptive horizon nonlinear model predictive control." *Journal of Process Control* 70 (2018): 109-122.
[4] P. Scokaert and D.Q. Mayne. "Min-max feedback model predictive control for constrained linear systems." *IEEE Transactions on Automatic control* 43.8 (1998): 1136-1142.
[5] K-U. Klatt and S. Engell. "Gain-scheduling trajectory control of a continuous stirred tank reactor." *Computers & Chemical Engineering* 22.4-5 (1998): 491-502.
[6] J. Andersson, J. Gillis, G. Hor, J.B. Rawlings and M. Diehl. "CasADi: a software framework for nonlinear optimization and optimal control." *Mathematical Programming Computation* 11.1 (2019): 1-36.
[7] A. Wächter and L.T. Biegler. "IPOPT-an interior point OPTimizer." (2009): 156.
[8] M. Maiworm, T. Bäthge and R. Findeisen. "Scenario-based model predictive control: Recursive feasibility and stability." *IFAC-PapersOnLine* 48.8 (2015): 50-56.
[9] D. Krishnamoothy, L.T. Biegler and J. Jäschke. "Adaptive horizon economic nonlinear model predictive control." *Journal of Process Control.* 92 (2020): 108-118.
[10] P.J. Campo, J. Peter and M. Morari. "Robust model predictive control." 1987 *American control conference*. IEEE, 1987.
[11] D. Krishnamoorthy, B. Foss and S. Skogestad. "A Primal decomposition algorithm for distributed multistage scenario model predictive control." *Journal of Process Control* 81 (2019): 162-171.
[12] S. Lucia. "Robust multi-stage nonlinear model predictive control." Shaker, 2014.
[13] Z.J. Yu and L.T. Biegler. "Advanced-step multistage nonlinear model predictive control: Robustness and stability." *Journal of Process Control* 84 (2019): 192-206.
[14] A.V. Fiacco. "Sensitivity analysis for nonlinear programming using penalty methods." *Mathematical programming* 10.1 (1976): 287-311.
[15] D.M. Raimondo, D. Limon, M. Lazar, L. Magni, E.F. ndez Camacho. "Min-max model predictive control of nonlinear systems: a unifying overview on stability". *European Journal of Control* 15.1 (2009): 5-21.
[16] R. Findeisen and F. Allgöwer. "Computational delay in nonlinear model predictive control". *Proc. Int. Symp. Adv. Contr. of Chem. Proc* (2004): 427-432.
[17] L. Grüne and J. Pannek. "Nonlinear model predictive control". in: *Nonlinear Model Predictive Control*. Springer, (2017) : 45–69.
[18] S. Lucia, S. Subramanian, D. Limon and S. Engell. "Stability properties of multi-stage nonlinear model predictive control." *Systems & Control Letters* 143 (2020): 104743