

Uncertainty Propagation by Linear Regression Kalman Filters for Stochastic NMPC

Quirynen, Rien; Berntorp, Karl

TR2021-084 July 12, 2021

Abstract

Stochastic nonlinear model predictive control (SNMPC) allows to directly take model uncertainty into account, e.g., by including probabilistic chance constraints. This paper proposes linear-regression Kalman filtering to perform high-accuracy propagation of mean and covariance information for the nonlinear system dynamics in a tractable approximation of the stochastic optimal control problem. In addition, a tailored adjoint-based sequential quadratic programming (SQP) algorithm is presented to considerably reduce the computational cost and allow a real-time implementation of the resulting SNMPC. The prediction accuracy and control performance of the proposed approach are illustrated on a vehicle control application subject to external disturbances, while highlighting a worst-case computation time of 10 ms for SNMPC which is close to that of deterministic NMPC for this particular case study.

IFAC Conference on Nonlinear Model Predictive Control

Uncertainty Propagation by Linear Regression Kalman Filters for Stochastic NMPC

Rien Quirynen* Karl Berntorp*

* Mitsubishi Electric Research Laboratories, Cambridge, MA, 02139,
USA (e-mail: {quirynen,berntorp}@merl.com).

Abstract: Stochastic nonlinear model predictive control (SNMPC) allows to directly take model uncertainty into account, e.g., by including probabilistic chance constraints. This paper proposes linear-regression Kalman filtering to perform high-accuracy propagation of mean and covariance information for the nonlinear system dynamics in a tractable approximation of the stochastic optimal control problem. In addition, a tailored adjoint-based sequential quadratic programming (SQP) algorithm is presented to considerably reduce the computational cost and allow a real-time implementation of the resulting SNMPC. The prediction accuracy and control performance of the proposed approach are illustrated on a vehicle control application subject to external disturbances, while highlighting a worst-case computation time of 10 ms for SNMPC which is close to that of deterministic NMPC for this particular case study.

Keywords: Stochastic nonlinear model predictive control, linear-regression Kalman filtering, sequential quadratic programming, real-time optimization algorithms.

1. INTRODUCTION

Nonlinear model predictive control (NMPC) has grown mature and algorithmic techniques exist to handle relatively complex constrained control systems (Rawlings et al., 2017). Although NMPC exhibits an inherent robustness due to the state feedback (Allan et al., 2017), it does not take uncertainties directly into account. One alternative approach is robust NMPC, which relies on the optimization of control policies under worst-case scenarios in the presence of bounded disturbances and/or uncertainty. Stochastic NMPC aims at reducing the conservativeness of robust NMPC by directly incorporating the probabilistic description of uncertainties into the optimal control problem (OCP) (Mesbah, 2016).

In the present paper, we aim to solve the following OCP at each time step of the stochastic NMPC (SNMPC)

$$\min_{\kappa(\cdot)} \frac{1}{2} \mathbb{E} \left(\sum_{k=0}^{N-1} \|r(x_k, \kappa(x_k))\|_2^2 + \|r_N(x_N)\|_2^2 \right) \quad (1a)$$

$$\text{s.t. } x_0 = \hat{x}_t, \quad (1b)$$

$$x_{k+1} = f(x_k, \kappa(x_k), w_k), \quad \forall k \in \mathbb{Z}_0^{N-1}, \quad (1c)$$

$$0 \geq g_j(x_k, \kappa(x_k)), \quad \forall j \in \mathbb{Z}_1^{n_c}, k \in \mathbb{Z}_0^N, \quad (1d)$$

$$\epsilon_j \geq \Pr(h_j(x_k, \kappa(x_k)) > 0), \forall j \in \mathbb{Z}_1^{n_h}, k \in \mathbb{Z}_0^N, \quad (1e)$$

where $x_k \in \mathbb{R}^{n_x}$ denotes the state, $w_k \in \mathbb{R}^{n_w}$ the process noise, $\kappa : \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_u}$ is the state feedback law and $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_w} \rightarrow \mathbb{R}^{n_x}$ denotes the nonlinear system dynamics. We assume a least-squares cost function in (1a) and the disturbance variables $w_k \sim \mathcal{N}(\bar{w}_k, \Sigma_k)$ are independent and normally distributed with mean \bar{w}_k and covariance Σ_k . Eq. (1e) denotes a set of *individual chance constraints*, i.e., the probability of violating the constraint $h_j(\cdot) \leq 0$ is below a specified threshold value $\epsilon_j > 0$.

The optimization over state feedback policies $\kappa(\cdot)$ and the chance constraints (1e) are computationally intractable and typically require an approximate formulation. Scenario-based SNMPC computes a closed-loop policy, using particular predictions of the stochastic disturbance sequences. However, choosing the number of scenarios leads to a trade off between robustness and efficiency (Schildbach et al., 2014). Polynomial chaos (PC)-based MPC replaces the implicit mappings with expansions of orthogonal polynomial basis functions (Fagiano and Khammash, 2012). For time-varying uncertainties, PC-based SNMPC requires a large number of expansion terms (Mesbah, 2016).

Alternatively, an uncertainty propagation can be used to approximate the chance constraints, e.g., see (Telen et al., 2015) based on the mean and covariance prediction equations from the extended Kalman filter (EKF) (Gustafsson and Hendeby, 2012). Recent work in (Feng et al., 2020) proposed a tailored Jacobian approximation in an adjoint-based sequential quadratic programming (SQP) algorithm to solve the resulting SNMPC problems at a computational cost that is close to that of deterministic NMPC. The work in (Hewing et al., 2020) uses EKF equations to approximate chance constraints, based on Gaussian-process (GP) regression for residual model uncertainty.

It is known that unscented Kalman filtering (UKF) is more accurate than the EKF-based propagation of mean and covariance information for nonlinear system dynamics (Julier and Uhlmann, 2004). Linear-regression Kalman filtering (LRKF), of which UKF is a special case, is based on *statistical linearization* instead of an explicit linearization based on a Taylor-series approximation in the EKF (Steinbring and Hanebeck, 2013). Therefore, the EKF is a first-order method to handle nonlinearities, while the family of LRKFs can achieve second or higher order of accuracy (Julier and Uhlmann, 2004). The accuracy

of mean and covariance predictions are important when approximating the stochastic OCP (SOCP) in (1) by a tractable nonlinear program (NLP).

This paper presents the first computationally tractable implementation of SNMPC using LRKF-based mean and covariance propagation techniques. We propose an LRKF-based SOCP formulation and present a novel extension of the adjoint-based SQP algorithm from (Feng et al., 2020) to solve it efficiently online. Based on numerical simulation results for a vehicle-control case study, we illustrate the increased accuracy and closed-loop control performance due to the LRKF-based uncertainty propagation. Finally, using the tailored Jacobian approximation in the adjoint-based SQP optimization algorithm, the real-time computational cost for the proposed SNMPC implementation is shown to be close to that of deterministic NMPC.

The paper is organized as follows. Section 2 summarizes the uncertainty prediction in Kalman filtering, and Section 3 introduces the EKF- and LRKF-based SNMPC formulations. Then, Section 4 presents the tailored adjoint-based SQP algorithm. Results of the case study are presented in Section 5 and Section 6 concludes the paper.

2. PRELIMINARIES ON KALMAN FILTERING

Let us consider the nonlinear system dynamics in (1c) at time step k of the following form

$$x_{k+1} = f(x_k, u_k, w_k), \quad (2)$$

where $u_k \in \mathbb{R}^{n_u}$ denotes the control inputs and the process noise $w_k \sim \mathcal{N}(\bar{w}_k, \Sigma_k)$ is assumed to be a normally distributed signal with mean \bar{w}_k and covariance Σ_k . In practice, the mean and covariance can be estimated online and may be time-varying in our SNMPC approach.

2.1 Extended Kalman Filtering (EKF)

A first computationally tractable approach to propagate mean and covariance information for the nonlinear system in (2) uses an explicit linearization based on a Taylor-series approximation for the mean disturbance (Telen et al., 2015). This results in the first-order approximation of the mean and state covariance propagation

$$s_{k+1} = f(s_k, u_k, \bar{w}_k), \quad s_0 = \hat{x}_t, \quad (3a)$$

$$P_{k+1} = A_k P_k A_k^\top + B_k \Sigma_k B_k^\top, \quad P_0 = \hat{P}_t, \quad (3b)$$

where the Jacobian matrices A_k and B_k read as

$$A_k = \frac{\partial f}{\partial x}(s_k, u_k, \bar{w}_k), \quad B_k = \frac{\partial f}{\partial w}(s_k, u_k, \bar{w}_k). \quad (4)$$

The matrix $P_k \in \mathbb{R}^{n_x \times n_x}$ denotes the covariance for the mean state value $s_k \in \mathbb{R}^{n_x}$ and \hat{P}_t is the uncertainty of the current state estimate \hat{x}_t in (1b). We adopt the discrete-time Lyapunov equations in (3b) instead of a continuous-time formulation (Telen et al., 2015), to reduce the computational cost and preserve the positive definiteness.

2.2 Linear-Regression Kalman Filtering (LRKF)

In Gaussian filters, of which LRKFs are a subset, the distribution of the state prediction at time step $k+1$ is approximated by a Gaussian

$$p(x_{k+1}|x_k) \approx \mathcal{N}(\mathbb{E}[x_{k+1}], \text{Cov}[x_{k+1}]), \quad (5)$$

based on the associated moment integrals. In general, no closed-form solutions exist for the moment integrals, but numerical integration methods or *cubature rules* can be used instead. In LRKFs, a sampling-based approximation is used for the mean $\mathbb{E}[x_{k+1}] \approx s_{k+1}$ and state covariance $\text{Cov}[x_{k+1}] \approx P_{k+1}$, see (Steinbring and Hanebeck, 2013).

Let us define coordinate transforms $\xi_x = L^{x-1}(x - s)$ and $\xi_w = L^{w-1}(w - \bar{w})$, given the Cholesky factorization of the state covariance matrix $P_k = L_k^x L_k^{x\top}$ and of the process noise covariance $\Sigma_k = L_k^w L_k^{w\top}$. LRKFs are then based on pairs of weights and integration points $\mathcal{P} = \{\omega^{(i)}, \xi^{(i)}\}_{i=1}^{|\mathcal{P}|}$, where $\xi^{(i)} = (\xi_x^{(i)}, \xi_w^{(i)})$ and $|\mathcal{P}|$ denotes the cardinality. Given the integration points $\xi^{(i)}$ for $i \in \mathbb{Z}_1^{|\mathcal{P}|}$, we evaluate the nonlinear state transition map (2) as

$$x_{k+1}^{(i)} = f\left(s_k + L_k^x \xi_x^{(i)}, u_k, \bar{w}_k + L_k^w \xi_w^{(i)}\right), \quad \forall i \in \mathbb{Z}_1^{|\mathcal{P}|}, \quad (6)$$

given the control action u_k and mean state value s_k . Given these evaluated points $x_{k+1}^{(i)}$, the moment integrals can be approximated by the finite sums

$$s_{k+1} = \sum_{i=1}^{|\mathcal{P}|} \omega^{(i)} x_{k+1}^{(i)}, \quad (7a)$$

$$P_{k+1} = \sum_{i=1}^{|\mathcal{P}|} \omega_c^{(i)} \left(x_{k+1}^{(i)} - s_{k+1}\right) \left(x_{k+1}^{(i)} - s_{k+1}\right)^\top, \quad (7b)$$

where $\omega_c^{(1)} = \omega^{(1)} + (1 - \gamma^2 + \beta)$ for the central integration point and $\omega_c^{(i)} = \omega^{(i)}$ for $i \neq 1$. Let us introduce the choice of integration points, weights, and parameter values for two well known LRKF implementations.

Definition 1. The spherical cubature (SC) rule, as used in the cubature Kalman filter (CKF) (Arasaratnam, 2009), defines a set of $|\mathcal{P}| = 2n = 2(n_x + n_w)$ integration points $(\xi_x^{(i)}, \xi_w^{(i)})$ and weights ω_i as follows

$$\Xi = \sqrt{n} [\mathbf{1}_n \quad -\mathbf{1}_n], \quad \Omega = \frac{1}{2n} \mathbf{1}_{2n}^\top, \quad (8)$$

where $\mathbf{1}_n$ denotes the $n \times n$ identity matrix and $\mathbf{1}_{2n}$ is a vector of $2n$ elements that are equal to one. The integration points and weights correspond to the columns of Ξ and the elements of Ω , respectively. Note that the SC rule does not include a central point, i.e., $\gamma = 1$ and $\beta = 0$.

Definition 2. The unscented transform (UT), as used in the UKF (Gustafsson and Hendeby, 2012), defines a set of $|\mathcal{P}| = 2n + 1$ integration points and weights as

$$\Xi = \sqrt{(n + \lambda)} [\mathbf{0}_n \quad \mathbf{1}_n \quad -\mathbf{1}_n], \quad \Omega = \frac{1}{\lambda + n} \left[\lambda \frac{1}{2} \mathbf{1}_{2n}^\top \right],$$

where $\lambda = \gamma^2(n + \kappa) - n$ and the UT includes a central integration point $\xi^{(1)} = \mathbf{0}_n$ for which the weight value $\omega_c^{(1)} = \omega^{(1)} + (1 - \gamma^2 + \beta)$ in (7b). Different parameter values can be used to choose the sigma points, see (Gustafsson and Hendeby, 2012), but we further use $\gamma = \sqrt{\frac{3}{n}}$, $\beta = \frac{3}{n} - 1$, and $\kappa = 0$ such that $\lambda = \gamma^2(n + \kappa) - n = 3 - n$.

2.3 Consistency and Prediction Accuracy

The notion of consistency is crucial for the use of estimates in closed-loop control. A mean state estimate s of a state variable x with covariance P is consistent if $\text{Cov}[x] = \mathbb{E}[(x - s)(x - s)^\top] \preceq P$, i.e., the estimate covariance P does not underestimate the true state covariance. When

an estimate becomes inconsistent, in closed-loop control, it means that the controller could rely too much on the state estimate, with potentially detrimental impact on the subsequent control actions. As the EKF performs explicit linearization, it does not take linearization errors into account, and it therefore provides inconsistent state estimates when the dynamics are not sufficiently well behaved or when the initial error is large.

On the other hand, the statistical linearization in LRFs can be interpreted as a least-squares minimization of the function values of the nonlinear and linearized function, and it is possible to quantify the magnitude of induced errors and degree of consistency (Lefebvre et al., 2004). Due to the improved consistency and higher prediction accuracy of LRFs compared to the EKF, the resulting closed-loop control is expected to be more robust. It is noted in (Wan and Van Der Merwe, 2000) that the UKF is correct to the second order, even for non-Gaussian inputs, and higher order accuracy is conditioned on particular parameter choices. For certain classes of nonlinear transformations, statistical linearization can be done analytically, which further improves performance (Greiff et al., 2020).

3. UNCERTAINTY PROPAGATION TECHNIQUES FOR STOCHASTIC NMPC

To arrive at a tractable approximation of the OCP in (1), for simplicity, let us define the control action as $u_k + K s_k$ based on control inputs $u_k \in \mathbb{R}^{n_u}$, mean state value $s_k \in \mathbb{R}^{n_x}$ and the feedback gain matrix $K \in \mathbb{R}^{n_u \times n_x}$ to prestabilize the nonlinear system dynamics. The techniques in this paper could be extended to alternative feedback parameterizations, e.g., see (Mesbah, 2016).

3.1 Probabilistic Chance Constraints

The individual chance constraints in (1e) ensure that the probability of violating each of the path constraints $h_j(x_k, u_k) \leq 0$ is below a certain probability level $\epsilon_j > 0$. Based on the propagation of the mean and covariance, each chance constraint can be approximated by

$$h_j(s_k, u_k) + c_j \sqrt{D_{k,j} P_k D_{k,j}^\top} \leq 0, \quad (9)$$

where $D_k = \frac{\partial h}{\partial x}(s_k, u_k)$ is the constraint Jacobian matrix and $D_{k,j}$ is the j^{th} row of D_k . The back-off coefficient value c_j is computed to ensure the probability level ϵ_j in the chance constraint (1e). One option is to use the Cantelli-Chebyshev inequality, i.e., $c_j = \sqrt{\frac{1-\epsilon_j}{\epsilon_j}}$, which holds regardless of the underlying probability distribution, but may lead to relatively conservative bounds (Telen et al., 2015). An alternative approach is based on the assumption of normally distributed state trajectories, such that the coefficient c_j can be chosen as

$$c_j = \sqrt{2} \operatorname{erf}^{-1}(1 - 2\epsilon_j), \quad (10)$$

where $\operatorname{erf}^{-1}(\cdot)$ is the inverse error function.

3.2 Stochastic OCP: Extended Kalman Filtering

The proposed approximate SOCP formulation, using the EKF-based uncertainty propagation in (3), reads as

$$\min_{s, u, L} \frac{1}{2} \sum_{k=0}^{N-1} \|r(s_k, u_k + K s_k)\|_2^2 + \|r_N(s_N)\|_2^2 \quad (11a)$$

$$\text{s.t. } s_0 = \hat{x}_t, \quad L_0^\times = \operatorname{chol}(\hat{P}_t), \quad (11b)$$

$$s_{k+1} = f(s_k, u_k + K s_k, \bar{w}_k), \quad \forall k \in \mathbb{Z}_0^{N-1}, \quad (11c)$$

$$L_{k+1}^\times = \operatorname{chol}\left(\tilde{A}_k L_k^\times L_k^{\times\top} \tilde{A}_k^\top + \tilde{B}_k \Sigma_k \tilde{B}_k^\top\right), \quad (11d)$$

$$0 \geq g_j(s_k, u_k + K s_k), \quad \forall j \in \mathbb{Z}_1^{n_c}, k \in \mathbb{Z}_0^N, \quad (11e)$$

$$0 \geq h_j(s_k, u_k + K s_k) + c_j \sqrt{D_{k,j} L_k^\times L_k^{\times\top} D_{k,j}^\top}, \quad (11f)$$

$$\forall j \in \mathbb{Z}_1^{n_h}, k \in \mathbb{Z}_0^N,$$

given the Jacobian matrices $\tilde{A}_k = \frac{\partial f}{\partial x}(s_k, u_k + K s_k, \bar{w}_k)$ and $\tilde{B}_k = \frac{\partial f}{\partial w}(s_k, u_k + K s_k, \bar{w}_k)$. The optimization variables include the Cholesky factors $L_k^\times \in \mathbb{R}^{\frac{n_x(n_x+1)}{2}}$ for $k \in \mathbb{Z}_1^N$. Unlike the OCP formulation in (Feng et al., 2020) using covariance matrix variables, our proposed SOCP in (11) uses the Cholesky factors directly such that the corresponding state covariance matrices $P_k = L_k^\times L_k^{\times\top} \succ 0$ are guaranteed to be positive definite at each iteration of the numerical optimization method.

Remark 1. The use of a Cholesky factorization in OCP (11) requires this operation to be defined everywhere. The matrix expression in (11d) is positive semi-definite by design and can be ensured to be positive definite by adding a small regularization term $\delta > 0$ as

$$L_{k+1}^\times = \operatorname{chol}\left(\tilde{A}_k L_k^\times L_k^{\times\top} \tilde{A}_k^\top + \tilde{B}_k \Sigma_k \tilde{B}_k^\top + \delta I\right). \quad (12)$$

First and higher-order derivatives of the Cholesky factorization can be computed by algorithmic differentiation (AD) tools, e.g., in CasADi (Andersson et al., 2018).

3.3 Stochastic OCP: Linear-Regression Kalman Filtering

Using the LRF-based uncertainty propagation in (7), the approximate SOCP formulation reads as in (11) but (11c)-(11d) are replaced by the following equations

$$s_{k+1} = \sum_{i=1}^{|\mathcal{P}|} \omega^{(i)} x_{k+1}^{(i)}, \quad \forall k \in \mathbb{Z}_0^{N-1}, \quad (13a)$$

$$L_{k+1}^\times = \operatorname{chol}(\mathcal{Y}_{k+1} \mathcal{Y}_{k+1}^\top), \quad \forall k \in \mathbb{Z}_0^{N-1}, \quad (13b)$$

where $\Sigma_k = L_k^w L_k^{w\top}$, $P_k = L_k^\times L_k^{\times\top} \succ 0$, and the expressions for $x_{k+1}^{(i)}$ and \mathcal{Y}_{k+1} , $\forall k \in \mathbb{Z}_0^{N-1}$ read as

$$x_{k+1}^{(i)} = \phi\left(s_k + L_k^\times \xi_x^{(i)}, u_k, \bar{w}_k + L_k^w \xi_w^{(i)}\right), \quad \forall i \in \mathbb{Z}_1^{|\mathcal{P}|}, \quad (14a)$$

$$\mathcal{Y}_{k+1, i} = \sqrt{\omega_c^{(i)}} \left(x_{k+1}^{(i)} - s_{k+1}\right), \quad \forall i \in \mathbb{Z}_1^{|\mathcal{P}|}, \quad (14b)$$

where $\mathcal{Y}_{k+1, i}$ denotes the i^{th} column of the matrix \mathcal{Y}_{k+1} . We introduced the compact notation $\phi(\cdot)$ for the pre-stabilized dynamics in (14a) as

$$\phi(s_k, u_k, w_k) = f(s_k, u_k + K s_k, w_k). \quad (15)$$

Similar to Remark 1, a small regularization term $\delta > 0$ should be added in (13b) to ensure the OCP to be well-defined. In case of the SC or the UT rule in Definition 1 and 2, the LRF-based uncertainty propagation is based on $|\mathcal{P}| = 2n$ and $|\mathcal{P}| = 2n+1$ samples, respectively, where $n = n_x + n_w$. Based on the discussion in Section 2, the LRF-based SOCP formulation using (13) is known to provide advantages over the EKF-based SOCP in (11), in terms of consistency and prediction accuracy.

4. ADJOINT-BASED SEQUENTIAL QUADRATIC PROGRAMMING FOR SNMPC

Let us write the stochastic LRKF-based OCP formulation in (11) using (13) compactly as

$$\min_{y,z} \frac{1}{2} \|L(y)\|_2^2 \quad (16a)$$

$$\text{s.t. } 0 = F(y, z), \quad 0 = E(y, z), \quad 0 \geq I(y, z), \quad (16b)$$

based on the following shorthand notation for the trajectories of optimization variables

$$\begin{aligned} y &= [u_0^\top, s_1^\top, u_1^\top, \dots, s_{N-1}^\top, u_{N-1}^\top, s_N^\top]^\top, \\ z &= [\text{vec}(L_1^x)^\top, \dots, \text{vec}(L_{N-1}^x)^\top, \text{vec}(L_N^x)^\top]^\top, \end{aligned} \quad (17)$$

and for the constraint functions

$$F(\cdot) = \begin{bmatrix} s_1 - \sum_{i=1}^{|\mathcal{P}|} \omega^{(i)} x_1^{(i)} \\ \vdots \\ s_N - \sum_{i=1}^{|\mathcal{P}|} \omega^{(i)} x_N^{(i)} \end{bmatrix}, E(\cdot) = \begin{bmatrix} L_1^x - \text{chol}(\mathcal{Y}_1 \mathcal{Y}_1^\top) \\ \vdots \\ L_N^x - \text{chol}(\mathcal{Y}_N \mathcal{Y}_N^\top) \end{bmatrix}, \quad (18)$$

where $x_k^{(i)}$ and \mathcal{Y}_k are defined as in (14a) and (14b), $I(y, z)$ denotes the inequality constraints in (11e)-(11f) and $L(y)$ defines the least squares cost in (11a). Note that the initial state variables can be eliminated based on the conditions $s_0 = \hat{x}_t$ and $L_0^x = \text{chol}(\hat{P}_t)$ in (11b). The Lagrangian function for the NLP in (16) reads as

$$\Lambda(\cdot) = \frac{1}{2} \|L(y)\|_2^2 + \lambda^\top F(y, z) + \mu^\top E(y, z) + \nu^\top I(y, z),$$

where λ and μ are the Lagrange multipliers for the equality and ν the multipliers for the inequality constraints.

Remark 2. The NLP in (16) can similarly represent the EKF-based OCP formulation in (11). However, the state dynamic equations $0 = F(y, z)$ depend on both y and z in (13a) for the LRKF-based OCP, unlike the equations $0 = F(y)$ for the EKF-based OCP formulation in (11c).

4.1 Exact Jacobian-Based SQP Algorithm for SNMPC

In an SQP algorithm for solving the NLP (16), given the solution guess (y^i, z^i) at each iteration i , the following quadratic program (QP) is solved

$$\begin{aligned} \min_{\Delta y, \Delta z} \quad & \frac{1}{2} \Delta y^{i\top} H^i \Delta y^i + g^{i\top} \Delta y^i \\ \text{s.t.} \quad & \begin{cases} \sigma_{\text{F}}^i \Big| 0 = \begin{bmatrix} F(y^i, z^i) \\ E(y^i, z^i) \end{bmatrix} + \begin{bmatrix} \frac{\partial F}{\partial y}(\cdot) & \frac{\partial F}{\partial z}(\cdot) \\ \frac{\partial E}{\partial y}(\cdot) & \frac{\partial E}{\partial z}(\cdot) \end{bmatrix} \begin{bmatrix} \Delta y^i \\ \Delta z^i \end{bmatrix}, \\ \sigma_{\text{I}}^i \Big| 0 \geq I(y^i, z^i) + \begin{bmatrix} \frac{\partial I}{\partial y}(\cdot) & \frac{\partial I}{\partial z}(\cdot) \end{bmatrix} \begin{bmatrix} \Delta y^i \\ \Delta z^i \end{bmatrix}, \end{cases} \end{aligned} \quad (19)$$

to compute the primal search direction $(\Delta y^i, \Delta z^i)$ and Lagrange multiplier values $(\sigma_{\text{F}}^i, \sigma_{\text{E}}^i, \sigma_{\text{I}}^i)$. The SQP method updates the iterates as $y^{i+1} \leftarrow y^i + \alpha^i \Delta y^i$ and $z^{i+1} \leftarrow z^i + \alpha^i \Delta z^i$, in which the step size α^i can be computed, e.g., using a line-search method. The multipliers are updated as $\lambda^{i+1} \leftarrow \lambda^i + \alpha^i (\sigma_{\text{F}}^i - \lambda^i)$, $\mu^{i+1} \leftarrow \mu^i + \alpha^i (\sigma_{\text{E}}^i - \mu^i)$, and $\nu^{i+1} \leftarrow \nu^i + \alpha^i (\sigma_{\text{I}}^i - \nu^i)$. Alternative techniques can be used to ensure global convergence such as in trust-region SQP methods (Nocedal and Wright, 2006).

Due to the least-squares form of the objective in (16a), the NLP can be solved by the generalized Gauss-Newton (GGN)

variant of SQP (Gros et al., 2020). In this case, the Hessian of the Lagrangian $\Lambda(\cdot)$ can be approximated as

$$H^i = \frac{\partial L}{\partial y}(y^i)^\top \frac{\partial L}{\partial y}(y^i) \approx \nabla_y^2 \Lambda(y^i, z^i, \lambda^i, \mu^i, \nu^i), \quad (20)$$

and the gradient is computed as $g^i = \frac{\partial L}{\partial y}(y^i)^\top L(y^i)$. The GGN Hessian approximation (20) is known to be increasingly accurate for smaller values of the residual function $L(\cdot)$ in (16a), which makes it popular for tracking-type NMPC formulations (Rawlings et al., 2017). In contrast to SQP applied to deterministic NMPC, each QP subproblem (19) additionally involves the Cholesky factors in z for SNMPC, resulting in a total number of $N(n_x + n_u) + N\left(\frac{n_x(n_x+1)}{2}\right)$ QP variables. The latter leads to a considerable increase in the computational cost of each SQP iteration, which is asymptotically equal to $\mathcal{O}(N(n_x^2 + n_u)^3)$ when using a sparsity exploiting QP solver (Quirynen and Di Cairano, 2020), compared to $\mathcal{O}(N(n_x + n_u)^3)$ for deterministic NMPC that involves only $N(n_x + n_u)$ variables. To remedy this, we present an extension of the tailored inexact Newton-type implementation of SQP for SNMPC in (Feng et al., 2020), aimed at achieving a computational cost for solving the LRKF-based SNMPC (16) that is comparable to the cost for deterministic NMPC.

4.2 Adjoint-Based SQP Algorithm for SNMPC

The adjoint-based SQP from (Feng et al., 2020) cannot be directly applied to our proposed LRKF-based SNMPC problem in (16), due to Remark 2. Instead, to solve the NLP in (16), we propose an adjoint-based SQP method that solves the following QP in each iteration

$$\begin{aligned} \min_{\Delta y} \quad & \frac{1}{2} \Delta y^{i\top} H^i \Delta y^i + g_a^{i\top} \Delta y^i \\ \text{s.t.} \quad & \begin{cases} \sigma_{\text{F}}^i \Big| 0 = \tilde{F}(y^i, z^i) + \frac{\partial F}{\partial y}(y^i, z^i) \Delta y^i, \\ \sigma_{\text{I}}^i \Big| 0 \geq \tilde{I}(y^i, z^i) + \frac{\partial I}{\partial y}(y^i, z^i) \Delta y^i, \end{cases} \end{aligned} \quad (21)$$

and the adjoint-based gradient correction reads as

$$g_a^i = g^i + [\mathbf{1} \ 0] \left(J_{\text{eq}}^i - \tilde{J}_{\text{eq}}^i \right)^\top \begin{bmatrix} \lambda^i \\ \mu^i \end{bmatrix} = g^i + \frac{\partial E}{\partial y}(\cdot)^\top \mu^i, \quad (22)$$

where $g^i = \frac{\partial L}{\partial y}(y^i)^\top L(y^i)$ denotes the objective gradient and using the following Jacobian approximation

$$\tilde{J}_{\text{eq}}^i = \begin{bmatrix} \frac{\partial F}{\partial y}(\cdot) & \frac{\partial F}{\partial z}(\cdot) \\ 0 & \frac{\partial E}{\partial z}(\cdot) \end{bmatrix} \approx \begin{bmatrix} \frac{\partial F}{\partial y}(\cdot) & \frac{\partial F}{\partial z}(\cdot) \\ \frac{\partial E}{\partial y}(\cdot) & \frac{\partial E}{\partial z}(\cdot) \end{bmatrix} = J_{\text{eq}}^i. \quad (23)$$

As desired, and similar to deterministic NMPC, each QP subproblem (21) for the adjoint-based SQP method involves $N(n_x + n_u)$ variables and requires a computational cost of $\mathcal{O}(N(n_x + n_u)^3)$ using a sparsity exploiting optimization algorithm as in (Quirynen and Di Cairano, 2020).

Based on the Jacobian approximation (23) and because the Jacobian matrix $\frac{\partial E}{\partial z}(\cdot)$ is invertible, due to the structure of equality constraints in (18), the variables Δz^i are eliminated numerically from the QP in (21). However, they can be computed using the following expansion step

$$\Delta z^i = -\frac{\partial E}{\partial z}(y^i, z^i)^{-1} E(y^i, z^i). \quad (24)$$

Given Eq. (24), the resulting QP subproblem (21) depends only on state and control variables in Δy , based on the following updated evaluation of the inequality constraints

$$\begin{aligned}\tilde{I}(y^i, z^i) &= I(y^i, z^i) + \frac{\partial I}{\partial z}(\cdot)\Delta z^i \\ &= I(y^i, z^i) - \frac{\partial I}{\partial z}(\cdot)\frac{\partial E}{\partial z}(\cdot)^{-1}E(y^i, z^i),\end{aligned}\quad (25)$$

and similar expressions for the updated evaluation of the equality constraints

$$\tilde{F}(y^i, z^i) = F(y^i, z^i) - \frac{\partial F}{\partial z}(\cdot)\frac{\partial E}{\partial z}(\cdot)^{-1}E(y^i, z^i). \quad (26)$$

After solving the QP in (21) to compute Δy^i , σ_{F}^i , and σ_{I}^i , the update to the Lagrange multiplier values corresponding to the covariance propagation equations can be computed as

$$\sigma_{\text{E}}^i = -\frac{\partial E}{\partial z}(\cdot)^{-\top} \left(\frac{\partial F}{\partial z}(\cdot)^{\top} \sigma_{\text{F}}^i + \frac{\partial I}{\partial z}(\cdot)^{\top} \sigma_{\text{I}}^i \right), \quad (27)$$

including a contribution from the equations $0 = F(y, z)$ for the LRFK-based SOCP in (16), due to Remark 2.

Remark 3. The computations of Δz^i , $\tilde{I}(\cdot)$, $\tilde{F}(\cdot)$, and σ_{E}^i in (24)-(27) can be performed efficiently by exploiting the block-structured sparsity of the matrices, without the need for any matrix factorization. For example, based on the definition of $E(\cdot)$ in (18) and given $\Delta z_0^i = 0$, $\tilde{I} = I - \frac{\partial I}{\partial z} \frac{\partial E}{\partial z}^{-1} E(\cdot)$ can be computed recursively as

$$\Delta z_k^i = -E_k - \frac{\partial E_k}{\partial z_{k-1}} \Delta z_{k-1}^i, \quad \tilde{I}_k = I_k + \frac{\partial I_k}{\partial z_k} \Delta z_k^i, \quad (28)$$

for $k \in \mathbb{Z}_1^N$, based on directional derivatives that can be computed efficiently using AD (Andersson et al., 2018), where $z_k = \text{vec}(L_k^x)$ and E_k, I_k denote the equality and inequality constraints of the OCP at stage k , respectively.

4.3 Merit Function and Line Search SQP

The real-time iteration (RTI) algorithm performs one full-step SQP iteration at each NMPC time step without any globalization, under the assumption that the sampling time is sufficiently small and the iterates remain within a local contraction region when shifting trajectories from one time step to the next (Diehl et al., 2005). However, due to the strongly nonlinear equations in the SOCP formulations (11) and (13), we propose to use a globalization technique to improve convergence behavior of the resulting closed-loop system. In this work, we use a line-search method (Nocedal and Wright, 2006) based on the exact ℓ_1 penalty function for the NLP in (16),

$$m(y, z; \rho) = \frac{1}{2} \|L(y)\|_2^2 + \rho \|F(y, z)\|_1 + \rho \|E(y, z)\|_1 + \rho \sum_j \max(I_j(y, z), \epsilon), \quad (29)$$

where $\rho > 0$ is the penalty value and $\epsilon \geq 0$ denotes the feasibility tolerance for the inequality constraints. Note that the ℓ_1 merit function in (29) is not differentiable but directional derivatives exist. As discussed in (Nocedal and Wright, 2006), the merit function (29) is exact, i.e., a local minimizer (y^*, z^*) to the NLP (16) is a local minimizer of $m(y, z; \rho)$ for a sufficiently large penalty value $\rho > \rho^*$.

Similar to (Nocedal and Wright, 2006, Theorem 18.2), it can be shown that the search direction $(\Delta y, \Delta z)$ for

the exact or inexact adjoint-based SQP method forms a descent direction for the merit function $m(\cdot)$ in (29) for a sufficiently large parameter value $\rho > 0$. The line search then computes a step size $\alpha^i \in (0, 1]$ for which the following sufficient decrease condition holds

$$m(y^i + \alpha^i \Delta y^i, z^i + \alpha^i \Delta z^i; \rho) \leq m(y^i, z^i; \rho) + \alpha^i \eta \left[\nabla_y m(y^i, z^i; \rho)^\top \nabla_z m(y^i, z^i; \rho)^\top \right] \begin{bmatrix} \Delta y^i \\ \Delta z^i \end{bmatrix}, \quad (30)$$

which is based on the Armijo condition for unconstrained optimization and where $\eta \in (0, 1)$.

4.4 Real-Time Iterations for Stochastic NMPC

Based on standard SQP convergence results (Nocedal and Wright, 2006), the proposed line-search SQP method converges to a local minimizer for the NLP in (16). However, in order to achieve a real-time feasible implementation of SNMPC, we propose an extension of the RTI method (Gros et al., 2020) based on a single iteration of the adjoint-based SQP method in Section 4.2 at each time step. The resulting approach is detailed in Algorithm 1.

Algorithm 1 Real-time Adjoint-based SQP for SNMPC

- 1: **Input:** Guess $(y^i, z^i, \lambda^i, \mu^i, \nu^i)$, and feedback gain K .
 - 2: **Prepare** QP subproblem (21):
 - 3: Compute block-diagonal Hessian H^i and g_a^i in (22).
 - 4: Evaluate block-sparse Jacobians $\frac{\partial F}{\partial y}(\cdot)$ and $\frac{\partial I}{\partial y}(\cdot)$, and $\tilde{I}(\cdot)$ and $\tilde{F}(\cdot)$ in (25)-(26), using Remark 3.
 - 5: **Solve** block-sparse QP (21):
 - 6: Receive current state estimate \hat{x}_t and \hat{P}_t .
 - 7: Solve QP in (21) to obtain $\Delta y^i, \sigma_{\text{F}}^i$ and σ_{I}^i .
 - 8: **Expand** solution Δz^i in (24) and σ_{E}^i in (27).
 - 9: **Search** for step size selection:
 - 10: Compute $\alpha^i \in (0, 1]$ such that Eq. (30) holds.
 - 11: $y^{i+1} \leftarrow y^i + \alpha^i \Delta y^i, z^{i+1} \leftarrow z^i + \alpha^i \Delta z^i, \lambda^{i+1} \leftarrow \lambda^i + \alpha^i \Delta \lambda^i, \mu^{i+1} \leftarrow \mu^i + \alpha^i \Delta \mu^i$ and $\nu^{i+1} \leftarrow \nu^i + \alpha^i \Delta \nu^i$.
 - 12: **Feedback:** send control $u^* = u_0^{i+1} + K \hat{x}_t$ to process.
 - 13: **Shift** to next time step, see (Gros et al., 2020).
 - 14: **Output:** New values $(y^{i+1}, z^{i+1}, \lambda^{i+1}, \mu^{i+1}, \nu^{i+1})$.
-

5. CASE STUDY: STOCHASTIC NMPC FOR VEHICLE CONTROL UNDER UNCERTAINTY

The case study involves a reference tracking SNMPC for vehicle control under external disturbances on the tire-wheel angle and on the tire-friction parameters. We present numerical simulation results based on a C code implementation of Algorithm 1 that uses AD code generation in CasADi (Andersson et al., 2018) in combination with the primal active-set method in PRESAS (Quirynen and Di Cairano, 2020) to solve each QP subproblem.

5.1 Vehicle Control Problem Formulation

Similar to (Berntorp et al., 2020), we use a single-track vehicle model that includes the position (p^x, p^y) , the longitudinal velocity v^x , lateral velocity v^y , yaw angle ψ and yaw rate $\dot{\psi}$ as states, i.e., $n_x = 6$. The inputs to the vehicle model are the front and rear wheel speeds ω_f, ω_r and the tire-wheel angle δ , i.e., $n_u = 3$. The single-track

model lumps together the left and right wheel on each axle, while roll and pitch dynamics are neglected. The slip angles α_i and slip ratios λ_i are defined as in (Berntorp et al., 2020), and the tire forces are computed with Pacejka’s Magic Formula,

$$\begin{aligned} F_i^x &= \mu_i^x F_i^z \sin(D_i^x \arctan(B_i^x(1 - E_i^x)\lambda_i + E_i^x \arctan(B_i^x \lambda_i))), \\ F_i^y &= \eta_i \mu_i^y F_i^z \sin(D_i^y \arctan(B_i^y(1 - E_i^y)\alpha_i + E_i^y \arctan(B_i^y \alpha_i))), \end{aligned}$$

where F_i^z denote the normal forces, μ_i^j , B_i^j , D_i^j and E_i^j , for $i \in \{f, r\}, j \in \{x, y\}$, are the parameter values corresponding to a snow-covered road.

To illustrate the performance of our proposed SNMPC in Algorithm 1, we consider the following uncertainty model in the nonlinear vehicle dynamics

$$\delta_d \sim \mathcal{N}(0, \Sigma_\delta), \quad \mu_i^j \sim \mathcal{N}(\bar{\mu}_i^j, \Sigma_{i,j}), \quad (31)$$

where δ_d is an external disturbance on the front-wheel steering angle $\delta_f = \delta + \delta_d$ and the friction coefficients are normally distributed with mean values $\bar{\mu}_i^j$ for $i \in \{f, r\}, j \in \{x, y\}$. We define the stage cost in (1a) as

$$\|r(\cdot)\|_2^2 = \|x_k - x_{\text{ref},k}\|_Q^2 + \|u_k - u_{\text{ref},k}\|_R^2, \quad (32)$$

corresponding to state and control reference tracking. We enforce individual chance constraints for the following time-varying inequalities $h(\cdot) \leq 0$ in the OCP (1),

$$p_k^y \leq \bar{p}_k^y \leq \underline{p}_k^y, \quad \delta_k \leq \bar{\delta}_k \leq \underline{\delta}_k, \quad \omega_{i,k} \leq \bar{\omega}_{i,k} \leq \underline{\omega}_{i,k}, \quad (33)$$

for $i \in \{f, r\}$ and $\epsilon = 0.1$ in (10). The state-dependent inequality constraints are reformulated using a slack variable, with an exact ℓ_1 penalty in the cost function (Berntorp et al., 2020), to ensure that a feasible solution exists for the OCP at each time step in closed-loop simulations.

5.2 Open-loop Optimal Control Problem

Figure 1 illustrates the solution trajectories for the lateral position of the vehicle based on the proposed SOCP in (11)-(13). To allow a fair comparison of the prediction accuracy, Figure 1 shows the open-loop simulation results for the mean value and the state covariance using either the EKF- or LRKF-based equations in (3) or (7), respectively, based on the same trajectory of control input values. In addition, similar to (Gustafsson and Hendeby, 2012), we use Monte-Carlo simulations with 5000 random disturbance realizations to validate the accuracy of the open-loop simulation results for the lateral position of the vehicle. Using the UT in Definition 2, the LRKF-UT approach performs better than the EKF in this example, both for the mean and covariance trajectories.

5.3 Closed-loop SNMPC Simulation Results

Next, we can validate the closed-loop performance of the proposed SNMPC algorithm using the closed-loop cost and constraint violation defined as follows

$$\text{Cost} = \sum_k (\|x_k - x_{\text{ref},k}\|_Q^2 + \|u_k - u_{\text{ref},k}\|_R^2), \quad (34)$$

$$\text{Violation} = \sum_k T_s ((p_k^y - \bar{p}_k^y)_+ + (\underline{p}_k^y - p_k^y)_+),$$

where $(\cdot)_+ = \max(\cdot, 0)$. Table 1 displays the mean and maximum values of both metrics in (34) for closed-loop simulation results of deterministic versus stochastic NMPC, in which the SQP method converges at each control time step to obtain accurate comparisons. The results are based on 500 realizations of the disturbances in (31)

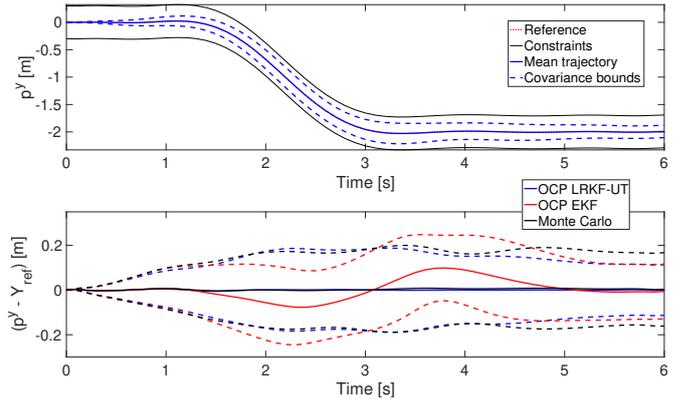


Fig. 1. Stochastic OCP for vehicle control: open-loop solution trajectory for lateral position (top) and reference tracking error for EKF and LRKF compared against 5000 Monte Carlo simulations (bottom).

and the vehicle maneuver is illustrated in Figure 1 on a snow-covered road surface at a reference speed of 12 m/s.

Table 1. Results for 500 disturbance realizations on snow at 12 m/s: closed-loop SNMPC simulations with converged SQP method.

Converged SQP	Cost		Violation	
	mean	max	mean	max
DETERMINISTIC NMPC	3.04e-01	1.65e+01	2.68e-03	2.46e-01
SNMPC: EKF	1.53e-01	3.54e+00	1.49e-03	1.44e-01
SNMPC: LRKF-SC	1.14e-01	2.27e+00	7.93e-04	1.83e-01
SNMPC: LRKF-UT	9.39e-02	1.01e+00	2.38e-04	5.38e-02

Table 1 includes results for SNMPC using the EKF-based SOCP in (11) and the LRKF-based SOCP in (13), using the SC and UT rule in Definition 1 and 2. The average and worst-case cost for SNMPC-EKF is a factor 2 and 4 times smaller than for deterministic NMPC, and the constraint violations are reduced greatly too. In addition, the LRKF-based SNMPC reduces both the cost and violations even further, while the LRKF-UT based SNMPC leads to the overall best performance. The SC-based LRKF reduces the mean cost with roughly 30 % compared to using EKF, and the UT-based LRKF improves even further. In addition, the constraint violations are heavily reduced when using the LRKF-based SNMPC formulations.

5.4 Real-time Feasibility of Adjoint-based SNMPC

Table 2 illustrates the real-time feasibility of the proposed adjoint-based SQP method in Algorithm 1, using one iteration at each control time step. The table shows average and worst-case computation times for 100 disturbance realizations, using either the adjoint or exact Jacobian SQP method for the LRKF- or EKF-based SNMPC. Table 2 presents timing results for the major computational steps including the preparation, the QP solution, and the line search. It can be observed that the proposed adjoint SQP method is at least 6 times faster than an exact Jacobian implementation. The worst-case computation times for adjoint-based SNMPC are very comparable to the worst-case computation time for deterministic NMPC. Specifi-

Table 2. Results for 100 disturbance realizations on snow at 12 m/s: average and worst-case computation times per sampling instant of RTI-based SNMPC simulations. ¹

	SNMPC: LRKF-UT		SNMPC: EKF		Deterministic NMPC
	Adjoint SQP	Exact SQP	Adjoint SQP	Exact SQP	Exact SQP
PREPARATION TIME (MS)	1.15/1.74	1.55/2.28	0.16/0.32	0.69/1.40	0.05/0.12
QP SOLUTION TIME (MS)	0.52/5.74	29.25/73.28	0.79/12.18	30.78/82.65	1.68/9.58
# OF QP ITERATIONS	20.6/206.0	922.0/1000.0	29.8/373.0	907.1/1000.0	77.8/393.0
LINE SEARCH TIME (MS)	0.55/2.04	0.50/1.96	0.03/0.19	0.06/0.46	0.05/0.19
# OF LS ITERATIONS	1.1/4.0	1.1/4.0	1.2/7.0	1.1/4.0	1.8/5.0
TOTAL CPU TIME (MS)	3.59/ 9.57	32.00/76.33	1.16/ 12.61	31.66/84.36	1.81/ 9.78

cally, the total computation time for the proposed LRKF-based SNMPC controller is below 10 ms and therefore well below a desirable sampling time of 50 ms for vehicle control applications (Berntorp et al., 2020).

6. CONCLUSION

This paper presents a novel approach for SNMPC with individual chance constraints, using LRKF to approximate the propagation of mean and covariance information for the nonlinear system dynamics in a computationally tractable formulation. An adjoint SQP optimization algorithm is presented with a tailored Jacobian approximation to result in a computational cost that is close to that of deterministic NMPC. A real-time feasible implementation of LRKF-based SNMPC is proposed, using only one adjoint SQP iteration per time step, and its performance is illustrated based on numerical simulation results for a vehicle control case study under external disturbances.

REFERENCES

- Allan, D.A., Bates, C.N., Risbeck, M.J., and Rawlings, J.B. (2017). On the inherent robustness of optimal and suboptimal nonlinear MPC. *Systems & Control Letters*, 106, 68 – 78.
- Andersson, J.A.E., Gillis, J., Horn, G., Rawlings, J.B., and Diehl, M. (2018). Casadi—a software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*.
- Arasaratnam, I. (2009). *Cubature Kalman filtering theory & applications*. Ph.D. thesis, McMaster University.
- Berntorp, K., Quirynen, R., Uno, T., and Di Cairano, S. (2020). Trajectory tracking for autonomous vehicles on varying road surfaces by friction-adaptive nonlinear model predictive control. *Vehicle System Dynamics*, 58(5), 705–725.
- Diehl, M., Bock, H.G., and Schlöder, J.P. (2005). A real-time iteration scheme for nonlinear optimization in optimal feedback control. *SIAM Journal on Control and Optimization*, 43(5), 1714–1736.
- Fagiano, L. and Khammash, M. (2012). Nonlinear stochastic model predictive control via regularized polynomial chaos expansions. In *IEEE Conference on Decision and Control*, 142–147.
- Feng, X., Di Cairano, S., and Quirynen, R. (2020). Inexact adjoint-based SQP algorithm for real-time stochastic nonlinear MPC. In *IFAC World Congress*.
- Greiff, M., Robertsson, A., and Berntorp, K. (2020). Exploiting linear substructure in linear regression Kalman filters. In *IEEE Conf. Decision and Control (CDC)*.
- Gros, S., Zanon, M., Quirynen, R., Bemporad, A., and Diehl, M. (2020). From linear to nonlinear MPC: bridging the gap via the real-time iteration. *International Journal of Control*, 93(1), 62–80.
- Gustafsson, F. and Hendeby, G. (2012). Some relations between extended and unscented Kalman filters. *IEEE Transactions on Signal Processing*, 60(2), 545–555.
- Hewing, L., Kabzan, J., and Zeilinger, M.N. (2020). Cautious model predictive control using gaussian process regression. *IEEE Transactions on Control Systems Technology*, 28(6), 2736–2743.
- Julier, S.J. and Uhlmann, J.K. (2004). Unscented filtering and nonlinear estimation. *Proc. IEEE*, 92(3), 401–422.
- Lefebvre, T., Bruyninckx, H., and De Schutter, J. (2004). Kalman filters for non-linear systems: a comparison of performance. *Int. J. Control*, 77(7), 639–653.
- Mesbah, A. (2016). Stochastic model predictive control: An overview and perspectives for future research. *IEEE Control Systems Magazine*, 36(6), 30–44.
- Nocedal, J. and Wright, S.J. (2006). *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer, 2 edition.
- Quirynen, R. and Di Cairano, S. (2020). PRESAS: Block-structured preconditioning of iterative solvers within a primal active-set method for fast model predictive control. *Optimal Control Applications and Methods*, 41(6), 2282–2307.
- Rawlings, J.B., Mayne, D.Q., and Diehl, M.M. (2017). *Model Predictive Control: Theory, Computation, and Design*. Nob Hill, 2nd edition.
- Schildbach, G., Fagiano, L., Frei, C., and Morari, M. (2014). The scenario approach for stochastic model predictive control with bounds on closed-loop constraint violations. *Automatica*, 50(12), 3009 – 3018.
- Steinbring, J. and Hanebeck, U.D. (2013). S2KF: The smart sampling Kalman filter. In *16th Int. Conf. Information Fusion*. Istanbul, Turkey.
- Telen, D., Vallerio, M., Cibanca, L., Houska, B., Van Impe, J., and Logist, F. (2015). Approximate robust optimization of nonlinear systems under parametric uncertainty and process noise. *Journal of Process Control*, 33, 140–154.
- Wan, E.A. and Van Der Merwe, R. (2000). The unscented Kalman filter for nonlinear estimation. In *Proc. IEEE Adaptive Systems for Signal Processing, Communications, and Control Symposium*, 153–158.

¹ Computation times were obtained on a modern computer that is equipped with a 2.4 GHz 8-Core Intel Core i9 processor.