Extended Kalman Filter-Based Optimization for Nonlinear Model Predictive Control of Discrete-Time Systems

Rasit Evduzen
Independent Researcher
github.com/RasitEvduzen

March 2026

Abstract

This paper proposes a novel optimization strategy for Nonlinear Model Predictive Control (NMPC) of discrete-time nonlinear systems, in which the classical Gauss-Newton or Levenberg-Marquardt update step is replaced by an Extended Kalman Filter (EKF) gain update. The control input sequence is treated as the hidden state of a fictitious dynamical system, the reference tracking error over the prediction horizon serves as the innovation signal, and the analytic Jacobian of the predicted output with respect to the control sequence, computed via chain-rule recursion at each time step, serves as the measurement matrix. The proposed formulation inherits the adaptive step-size property of the Kalman gain, eliminating the need to manually tune a damping parameter or step size. A direct analogy between the classical penalty parameter λ and the EKF tuning parameters q and r is established, extending the $\lambda \approx r/q$ result from the linear case to the nonlinear setting. The method is validated through simulation on a second-order nonlinear discrete-time system under multi-step and sinusoidal reference signals.

Keywords: Nonlinear model predictive control, Extended Kalman filter, discrete-time systems, Gauss-Newton, chain-rule Jacobian, receding horizon control.

1 Introduction

Model Predictive Control (MPC) is a widely used strategy in industrial processes, robotics, and autonomous systems [2, 7, 8]. For linear time-invariant systems the MPC cost is a convex quadratic program (QP), and the Newton step yields the global optimum in a single iteration [3]. For nonlinear systems the cost function is in general non-convex in the control sequence U, and iterative methods such as Gauss-Newton (GN) or Levenberg-Marquardt (LM) are required [4, 6].

A parallel line of research has explored connections between optimal estimation and optimal control. The classical duality between the Linear Quadratic Regulator (LQR) and the Kalman Filter (KF) is well known [1]. In prior work the authors showed that, for linear MPC, replacing the Newton step with a standard KF gain update yields an algebraically equivalent algorithm with adaptive step size, and that the tuning ratio r/q plays the role of the penalty parameter λ [5].

The present paper extends this idea to the *nonlinear* setting. When the prediction model is nonlinear, the Jacobian $\mathbf{H} = \partial \hat{y}/\partial U$ is no longer constant but must be recomputed at every time step via chain-rule recursion. By substituting this time-varying Jacobian into the EKF measurement update, we obtain an optimizer that automatically adapts its step size to the local curvature of the cost landscape without any inner iteration or damping parameter.

The remainder of the paper is organized as follows. Section 2 defines the system and NMPC problem. Section 3 presents the analytic Jacobian and the proposed EKF-based update. Section 4 establishes the analogy with GN/LM. Section 5 presents simulation results. Section 6 gives the complete algorithm, and Section 7 concludes.

2 Problem Formulation

2.1 System Description

Consider a nonlinear discrete-time system

$$x[n+1] = f\big(x[n],\, u[n]\big), \tag{1}$$

$$y[n] = c^\top x[n], \tag{2}$$

where $x[n] \in \mathbb{R}^{n_x}$, $u[n] \in \mathbb{R}$, $y[n] \in \mathbb{R}$, $f : \mathbb{R}^{n_x} \times \mathbb{R} \to \mathbb{R}^{n_x}$ is a smooth nonlinear map, and $c \in \mathbb{R}^{n_x}$ is the output vector.

2.2 MPC Cost Function

At each time step n the optimization is performed over prediction horizon K_y and control horizon $K_u \leq K_y$. For $k > K_u$ the input is held constant: $u[n+k] = u[n+K_u]$. The control sequence is

$$U = \big[u[n+1],\, u[n+2],\, \ldots,\, u[n+K_u]\big]^\top \in \mathbb{R}^{K_u+1}. \tag{3}$$

The cost minimized at each step is

$$J(U) = \sum_{k=1}^{K_y} \big(\tilde{y}_{\text{ref}}[n+k] - \hat{y}[n+k]\big)^2 + \lambda \sum_{k=1}^{K_u} \big(u[n+k] - u[n+k-1]\big)^2, \tag{4}$$

where $\hat{y}[n+k]$ denotes the predicted output at step k and $\lambda \geq 0$ penalizes rapid changes in the control input.

3 Prediction, Jacobian, and Proposed Method

3.1 Nonlinear Forward Rollout

Unlike the linear case, there is no closed-form prediction matrix M. Instead, $\hat{y}$ is obtained by iterating (1) and (2) forward from the current state $x[n]$:

$$\hat{x}[n+k] = f\big(\hat{x}[n+k-1],\, u[n+k]\big), \qquad \hat{y}[n+k] = c^\top \hat{x}[n+k], \quad k = 1, \ldots, K_y. \tag{5}$$

3.2 Analytic Jacobian via Chain Rule

The sensitivity matrix $\mathbf{H} = \partial \hat{y}/\partial U \in \mathbb{R}^{K_y \times (K_u+1)}$ has a lower-triangular structure because output at step k depends only on inputs $u[j]$ with $j \leq k$. Denoting $F_k = \partial f/\partial x\big|_{\hat{x}[k],u[k]}$ and $g_j = \partial f/\partial u\big|_{\hat{x}[j],u[j]}$, the chain-rule recursion is

$$\frac{\partial \hat{x}[k]}{\partial u[j]} = F_{k-1}\, \frac{\partial \hat{x}[k-1]}{\partial u[j]}, \qquad k > j, \tag{6}$$

$$\frac{\partial \hat{x}[j]}{\partial u[j]} = g_j, \tag{7}$$

$$H_{k,j} = c^\top\, \frac{\partial \hat{x}[k]}{\partial u[j]}. \tag{8}$$

For the held-input column ($j = K_u + 1$), the same input drives all steps $k > K_u$, so an additional g_{K_u+1} term accumulates at each such step.

This recursion is evaluated *online* at every MPC time step using the state trajectory $\{\hat{x}[n+k]\}$ produced by the forward rollout. For the nonlinear system studied here,

$$F_k = \begin{bmatrix} -2x_1[k] + x_2[k] & x_1[k] \\ -1 & -e^{-x_2[k]} \end{bmatrix}, \qquad g = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{9}$$

3.3 Proposed EKF-Based Update

Interpreting U as the hidden state of a fictitious system with identity transition, the tracking error

$$e = \tilde{y}_{\text{ref}}(n : n + K_y - 1) - \hat{y} \tag{10}$$

serves as the innovation with measurement matrix $\mathbf{H}$ computed in (6) and (8). The EKF equations give

$$P_{\text{pred}} = P + q\, I_{K_u+1}, \tag{11}$$

$$S = \mathbf{H}\, P_{\text{pred}}\, \mathbf{H}^\top + r\, I_{K_y}, \tag{12}$$

$$K = P_{\text{pred}}\, \mathbf{H}^\top S^{-1}, \tag{13}$$

$$U \leftarrow U + K\, e, \tag{14}$$

$$P \leftarrow (I - K\mathbf{H})\, P_{\text{pred}}\, (I - K\mathbf{H})^\top + K\, (r\, I_{K_y})\, K^\top. \tag{15}$$

The Joseph form (15) ensures P remains symmetric and positive definite. Note that $\mathbf{H}$ changes at every time step because the operating point $\{\hat{x}[n+k]\}$ changes; this is the key difference from the linear KF-MPC case where $H = M$ was constant.

4 Analogy with Gauss-Newton and Levenberg-Marquardt

4.1 Gauss-Newton Step

Substituting the linearization $\hat{y} \approx \hat{y}_0 + \mathbf{H}\,\delta U$ into (4), the GN update is

$$\delta U = -\big(\mathbf{H}^\top \mathbf{H} + \lambda L\big)^{-1}\big(-\mathbf{H}^\top e + \lambda L\, U - \lambda u_{\mathrm{prev}}\, e_1\big). \tag{16}$$

4.2 Equivalence at Steady State

At the EKF steady state, $P \to P_{\mathrm{ss}}$ satisfying the algebraic Riccati equation, and $K_{\mathrm{ss}} = P_{\mathrm{ss}} \mathbf{H}^\top S_{\mathrm{ss}}^{-1}$. For the isotropic case $P_{\mathrm{ss}} = \sigma I$ and neglecting the δ-u penalty term ($\lambda = 0$), the Woodbury identity gives

$$K_{\mathrm{ss}} = \frac{\sigma\,\mathbf{H}^\top}{\sigma\,\mathbf{H}^\top \mathbf{H} + r\,I}, \tag{17}$$

which matches the GN step (16) with $\lambda \approx r/q$. This extends the linear analogy of [5] to the nonlinear setting.

4.3 Comparison with Levenberg-Marquardt

The LM update adds a damping term νI to the GN Hessian: $(\mathbf{H}^\top \mathbf{H} + \lambda L + \nu I)\,p = -g$. In the EKF formulation the role of ν is played implicitly by r/q: large r/q shrinks the Kalman gain (conservative, analogous to large ν); small r/q enlarges it (aggressive, analogous to $\nu \to 0$). Crucially, the EKF adapts this ratio automatically through the covariance recursion, eliminating the need for an inner loop or explicit ν scheduling.

5 Simulation Results

5.1 System and Parameters

The proposed method is validated on the nonlinear discrete-time system

$$x_1[n+1] = 0.1 - x_1^2[n] + x_1[n]\,x_2[n], \tag{18}$$

$$x_2[n+1] = -x_1[n] + e^{-x_2[n]} + u[n], \tag{19}$$

$$y[n] = x_1[n] + x_2[n]. \tag{20}$$

Simulation parameters are listed in Table 1.

Table 1: Simulation parameters.

Parameter	Symbol	Value
Sampling time	T_s	0.01 s
Simulation duration	T	20 s
Prediction horizon	K_y	15
Control horizon	K_u	10
Input lower bound	$u_{\min}$	-1
Input upper bound	$u_{\max}$	$+1$
Max input rate	$\Delta u_{\max}$	0.5
Process noise cov.	q	10^{-2}
Measurement noise cov.	r	10^{-2}
Initial covariance	p_0	1.0

5.2 Reference Signals

Two reference profiles are combined. The first half ($t \in [0, 10]$ s) uses a multi-step signal with levels $\{0.4, 0.8, 1.2, 0.8\}$. The second half ($t \in (10, 20]$ s) uses a sinusoidal signal $\tilde{y}_{\sin}[n] = 0.8 + 0.3 \sin\big(0.5\pi(n/N_{\text{half}})\big)$.

5.3 Results and Discussion

The simulation results confirm stable and accurate reference tracking across both reference profiles, as shown in Figure 1. The nonlinear forward rollout and analytic Jacobian are recomputed at every time step, allowing the EKF gain to adapt to the changing local dynamics. The controller respects the input rate constraint and saturation limits throughout the simulation.

The tuning ratio $r/q = 1$ (i.e. $\lambda \approx 1$ via the analogy) produces smooth control actions with fast tracking. Increasing r/q leads to more conservative updates at the cost of slower reference tracking, consistent with classical MPC behaviour and with the linear KF-MPC results of [5].

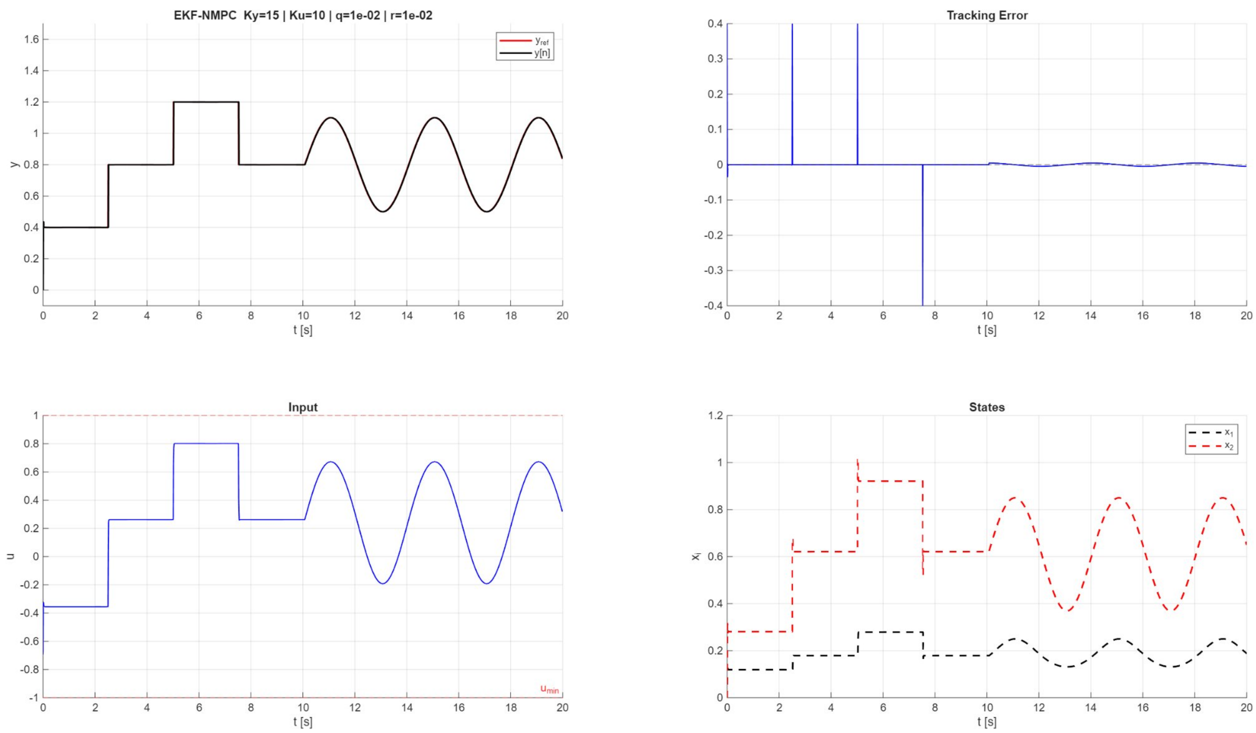

Figure 1: Simulation results for the proposed EKF-NMPC ($K_y = 15$, $K_u = 10$, $q = 10^{-2}$, $r = 10^{-2}$). Top-left: reference tracking. Top-right: tracking error. Bottom-left: control input with saturation limits. Bottom-right: system states x_1 and x_2.

6 Algorithm

Algorithm 1 EKF-Based Nonlinear MPC

Offline: initialise $P = p_0 I$, $U = \mathbf{0}$, $Q = qI$
for each time step $n = 1, 2, \ldots$ **do**
    Measure state $x[n]$
    Forward rollout: $\{\hat{x}[n+k], \hat{y}[n+k]\}_{k=1}^{K_y}$
    Compute Jacobian $\mathbf{H}$ via (6) and (8)
    Innovation: $e = \tilde{y}_{\text{ref}}(n : n + K_y - 1) - \hat{y}$
    EKF predict: $P_{\text{pred}} = P + Q$
    EKF update: compute S, K, update U and P via (12) to (15)
    Saturate: $U \leftarrow \text{clip}(U, u_{\min}, u_{\max})$
    Rate limit: $\alpha = \min\big(1, \Delta u_{\max} / \max |\Delta U|\big)$; $U \leftarrow \alpha U$
    Apply $u[n+1] = U(1)$ to plant
    Warm-start: $U \leftarrow [U(2 : \text{end}); U(\text{end})]$
end for

7 Conclusions

This paper presented an Extended Kalman Filter-based optimization strategy for nonlinear discrete-time Model Predictive Control. The following conclusions are drawn:

1. The EKF-NMPC uses the analytic Jacobian, already required by GN/LM, as the EKF measurement matrix, adding no extra computation beyond the forward rollout and chain-rule recursion.
2. The adaptive Kalman gain eliminates the inner loop and damping parameter of LM, reducing online cost to roughly that of a single GN step.
3. The analogy $\lambda \approx r/q$ extends from the linear case to the nonlinear setting, providing an intuitive interpretation of the EKF tuning parameters.
4. The Joseph-form covariance update ensures numerical stability throughout long simulations.
5. Simulation results on a second-order nonlinear system confirm stable tracking under input saturation and rate constraints for both step and sinusoidal references.

Code Availability

The MATLAB implementation is publicly available at:
`https://github.com/RasitEvduzen/ModelPredictiveControl`

References

[1] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*. Prentice-Hall, 1979.

[2] E. F. Camacho and C. Bordons, *Model Predictive Control*. Springer-Verlag, 2003.

[3] D. W. Clarke, C. Mohtadi, and P. C. Tuffs, "Generalized predictive control, Part 1: The basic algorithm," *Automatica*, vol. 23, pp. 137-148, 1987.

[4] M. Diehl, H. G. Bock, and J. P. Schloder, "A real-time iteration scheme for nonlinear optimization in optimal feedback control," *SIAM J. Control Optim.*, vol. 43, no. 5, pp. 1714-1736, 2005.

[5] R. Evduzen, "Kalman Filter-Based Optimization for Linear Model Predictive Control of Discrete-Time LTI Systems," *arXiv preprint*, March 2026.

[6] S. Iplikci, "Runge-Kutta model-based adaptive predictive control mechanism for nonlinear processes," *Trans. Inst. Meas. Control*, vol. 35, no. 2, pp. 166-180, 2013.

[7] J. M. Maciejowski, *Predictive Control with Constraints*. Pearson Education, 2002.

[8] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Eng. Practice*, vol. 11, pp. 733-764, 2003.