On the coupling of model predictive control and robust Kalman filtering

Alberto Zenere1, Mattia Zorzi2

1 Division of Automatic Control, Department of Electrical Engineering, Linköping University, SE-58183, Linköping, Sweden
2 Dipartimento di Ingegneria dell’Informazione, University of Padova, via Gradenigo 6/B, 35131 Padova, Italy
E-mail: zorzimat@dei.unipd.it

Abstract: Model predictive control (MPC) represents nowadays one of the main methods employed for process control in industry. Its strong suits comprise a simple algorithm based on a straightforward formulation and the flexibility to deal with constraints. On the other hand, it can be questioned its robustness regarding model uncertainties and external noises. Thus, a lot of efforts have been spent in the past years into the search of methods to address these shortcomings. In this study, the authors propose a robust MPC controller which stems from the idea of adding robustness in the prediction phase of the algorithm while leaving the core of MPC untouched. More precisely, they consider a robust Kalman filter that has been recently introduced and they further extend its usability to feedback control systems. Overall the proposed control algorithm allows to maintain all of the advantages of MPC with an additional improvement in performance and without any drawbacks in terms of computational complexity. To test the actual reliability of the algorithm, they apply it to control a servomechanism system characterised by non-linear dynamics.

1 Introduction

Model predictive control (MPC), also referred to as receding horizon control, is widely adopted in industry and represents the technology of choice to deal with the majority of constrained control problems [1–5]. MPC stems from the idea of employing a model that explains the relations among the variables of the plant to be controlled, which is then used to predict the future outputs. At each sampling time, a cost function $J_t$ comprising the tracking error (namely the difference between the desired trajectory and the predicted output) and the actuator efforts is minimised. The result of this procedure is an optimal sequence of future control moves which is applied according to a receding horizon philosophy: at time $t$ only the first input of the optimal command sequence is actually applied to the plant. The remaining optimal inputs are discarded, and a new minimisation control problem is solved at time $t+1$.

It is important to keep in mind that the employed model is always an approximation of the actual process and the more accurate the approximation is, the better the MPC performance is. On the other hand, there exists a trade-off between model accuracy and complexity of the optimisation: the simpler the model is (and thus less accurate), the easier is solving the optimisation problem. Therefore, the idea is to construct a prediction model that is very simple but at the same time representative enough to capture the main dynamical relations. Accordingly, a fundamental question about MPC regards its robustness with respect to model uncertainty and noise. Several different approaches can be found in literature to take into account model uncertainties when determining the optimal control sequence [6–14].

In practice, however, almost none of the methods presented is adopted and it is preferred instead an ad hoc MPC tuning [15, Section 1; 6, Section 10]. It is crucial to keep in mind that the user is looking for a controller that could actually be implemented in practice. It is then important to provide a control system that is feasible in terms of computational complexity and it is also easy to implement, even without an in-depth knowledge of the mathematics behind it.

This paper follows exactly this direction. We propose a robust MPC hinging on the fundamental observation that the predictions used in MPC heavily rely on the accuracy of the employed state-space model. Hence, the idea is to consider the usual MPC equipped with a robust Kalman filter whose predictions take into account the fact that the employed model is just an approximation of an accurate but complex (known or unknown) model of the actual process. More specifically in this paper we explore the use of the robust Kalman filter proposed in [16], see also [17–20]. According to this approach, all possible incremental models belong to a ball which is formed by placing a bound on the Kullback–Leibler divergence (or possibly other divergences [21, 22]) between the actual and the nominal incremental model. This bound represents the tolerance accounting the mismodelling budget. Then, the robust filter is obtained by minimising the mean square error according to the least favourable model in this ball.

It is important to note that this robust Kalman filter was developed in the context of systems without inputs [16, Section III]. Accordingly, in this paper we consider this robust filter to the more general case of controlled feedback systems. Remarkably, the filter still admits a Kalman-like structure leading to a simple implementation of the corresponding MPC algorithm and allowing a reasonably low computational burden. To test the effectiveness of the robust MPC, we apply it to control a servomechanism system characterised by non-linear dynamics. In particular, we compare its performance with standard MPC (i.e. MPC equipped with the Kalman filter) and MPC equipped with the risk-sensitive Kalman filter [23–26].

The outline of this paper is as follows. In Section 2, we briefly review classic MPC formulation. In Section 3, we present the main concepts behind the considered robust Kalman filter as well as its extension to systems with feedback and thus its applicability combined with MPC. In Section 4, we introduce the physics of the servomechanism system that we considered and we show different simulations to attest the efficiency of the control system, lastly in Section 5 the conclusions are drawn.

2 MPC formulation

2.1 Standard MPC

We briefly review the usual MPC formulation [27, Chapter 2]. We consider the discrete-time state-space model
respectively, and computed by using the Kalman filter

\begin{equation}
\Sigma : \begin{bmatrix} x_{t+1} = Ax_t + Bu_t + Gv_t \\ y_t = Cx_t + Dw_t \end{bmatrix} \tag{1}
\end{equation}

where \( x_t \in \mathbb{R}^n \), \( u_t \in \mathbb{R}^p \), \( y_t \in \mathbb{R}^q \) and \( v_t \in \mathbb{R}^r \) denote the state, the input, the output and unmeasured noise, respectively. We assume that \( x_t \) is Gaussian distributed with mean \( \hat{x}_t \) and covariance matrix \( P_t \) positive definite. For simplicity, in what follows, we assume that \( GD^T = 0 \), that is the state noise and the observation noise are independent. Assume that our task is for the output \( y_t \) to follow a certain reference signal \( r_t \). To this end, we introduce the following quadratic cost function:

\begin{equation}
J_t(u_t, \Sigma) = \sum_{k=1}^{H_0} \| \hat{y}_{t+k} - r_{t+k} \|_2^2 + \sum_{k=0}^{H_u-1} \| \Delta \hat{u}_{t+k} \|_2^2 \tag{2}
\end{equation}

Here \( \hat{y}_{t+k} \) represents the prediction of \( y_{t+k} \) at time \( t \) with \( k > 0 \);
\( \Delta \hat{u}_{t+k} := \hat{u}_{t+k} - \hat{u}_{t+k-1} \) is the predicted variation of the input from time \( t+k-1 \) to \( t+k \) and \( u_t := [u_t^T, u_{t+1}^T, \ldots] \). The prediction and control horizons have length \( H_P \) and \( H_u \) respectively, and \( H_u \leq H_P \). Lastly, \( Q_k \in \mathbb{R}^{p \times p} \) and \( R_k \in \mathbb{R}^{q \times q} \) denote the weight matrices for the output prediction errors at time \( t+k \) and for the predicted variations of the input at time \( t+k \). According to the receding horizon strategy, the control input \( u_{t+k} \) to apply to \( \Sigma \) at time \( t \) is extracted from \( u_t \) which is solution to the following open-loop optimisation problem:

\begin{equation}
u_{t+k} := \arg\min_{u_t} J_t(u_t, \Sigma) \tag{3}
\end{equation}

The well-known solution is [27, Chapter 3]

\begin{equation}u_t = [I_q 0 \cdots |(\Theta^T Q^2 + R)^{-1} \Theta^T Q (r_t - \Psi \hat{y}_t)] \tag{4}
\end{equation}

where

\begin{equation}\Psi = \begin{bmatrix} (C A)^T & (C A)^T & \cdots & (C A)^T \\ 0_{n \times q} & 0_{n \times q} & \cdots & 0_{n \times q} \\ C B & C B & \cdots & C B \\ \vdots & \vdots & \ddots & \vdots \\ C A^{H_0-1} B & C A^{H_0-H_1-1} B \\ Q = \text{diag}(Q_1, \ldots, Q_{H_0}) \\ R = \text{diag}(R_1, \ldots, R_{H_1-1}) \\ r_t = [r_{t+1}^T, r_{t+2}^T, \ldots, r_{t+H_1}^T]^T \end{bmatrix}
\end{equation}

and \( \hat{y}_t \) denotes the estimate of \( y_t \) at time \( t \). The latter is typically computed using the Kalman filter

\begin{align}
\hat{x}_{t+1} &= \hat{x}_{t+1} + L_t (y_t - C\hat{x}_{t+1}) \tag{5} \\
\hat{x}_{t+1} &= A\hat{x}_{t+1} + K_t (y_t - C\hat{x}_{t+1}) + B_u \tag{6} \\
L_t &= P_tC^T (C P_C^T + D D^T)^{-1}, \quad K_t = A L_t \tag{7} \\
P_{t+1} &= AP_a + K_t (C P_a C^T + D D^T) K_t^T + G G^T \tag{8}
\end{align}

where the initial conditions are \( \hat{x}_0 = \hat{y}_0 \) and \( P_0 = \hat{P}_0 \). The resulting MPC law is outlined by Algorithm 1.

Algorithm 1: [MPC law]

1: Collect the new data \( y_t \)
2: \( L_t = P_tC^T (C P_C^T + D D^T)^{-1} \)
3: \( \hat{x}_{t+t} = \hat{x}_{t+t} + L_t (y_t - C\hat{x}_{t+t}) \)
4: \( u_{t+1} = [I_q 0 \cdots |(\Theta^T Q^2 + R)^{-1} \Theta^T Q (r_t - \Psi \hat{y}_t)] \)
5: Apply \( u_{t+1} \) to the system
6: \( K_t = A L_t \)
7: \( P_{t+1} = AP_a + K_t (C P_a C^T + D D^T) K_t^T + G G^T \)
8: \( \hat{x}_{t+t} = A\hat{x}_{t+t} + K_t (y_t - C\hat{x}_{t+t}) + B_u \)
9: \( t \leftarrow t + 1 \)
10: Go to 1

Note that the estimator \( \hat{x}_t \) is computed by assuming to know the actual underlying model \( \Sigma \). In Section 2.2, we address the situation in which the actual model is different from \( \Sigma \).

It is worth noting that the optimisation problem (3) is usually considered with constraints, such as model uncertainty constraints and stability constraints. Such constraints are relatively easy to embed in (3), on the other hand the price to pay is that the corresponding problem does not admit a closed-form solution. Accordingly, the computational burden of Step 4 in Algorithm 1 increases. In what follows, we shall continue to consider the unconstrained MPC because, as we will see, we will embed the model uncertainty in a different way.

2.2 Robust MPC

Assume that the actual model, denoted by \( \tilde{\Sigma} \), is unknown and different from the nominal one, denoted by \( \Sigma \). It is then reasonable to assume that we are able to describe this uncertainty, that is we can characterise a set of models \( \delta \) for which \( \Sigma \in \delta \). In the robust MPC formulation, the optimisation problem (3) is usually substituted by the mini-max problem [6, Section 6]

\begin{equation}u_{t+1} = \arg\min_{u_t} \max_{\Sigma \in \delta} J_t(u_t, \Sigma) \tag{9}
\end{equation}

The latter is sometimes rewritten as a constrained MPC problem. It is worth noting that solving a mini-max problem (or a constrained MPC problem) is computationally more demanding than solving a min problem. Many different uncertainty descriptions have been proposed in the literature such as impulse response uncertainty [12], structured feedback uncertainty [10], polytopic uncertainty [28], disturbances uncertainty [13], probabilistic uncertainty [29] and Gaussian processes for modelling the underlying system [8]. Finally, in [30] it has been proposed a robust MPC wherein the cost function is an exponential-quadratic cost over the state distribution. In this way, large errors are severely penalised.

3 Robust MPC proposed

As we already noticed, the standard MPC relies on the assumption that the actual underlying model is known and thus the Kalman filter (5)-(8) is designed on it. This assumption, therefore, could deteriorate the performance of MPC when the actual model is different from the nominal one. We propose a robust MPC which stems from the idea of building a control system consisting of MPC on one hand, but equipped with a robust state estimator, that takes into account possible differences between the actual and the nominal model, on the other. In contrast with the usual robust MPC formulation, which is typically based on the mini-max Problem (9), we consider two independent optimisation problems:

• Robust estimation problem: we want to find a robust estimate \( \hat{x}_t \) of \( x_t \), independently of the the fact that it will be next used to determine the optimal control input \( u_{t+1} \). As we will see in Section 3.1, this problem is a mini–max problem itself, but its solution gives a robust filter obeying a Kalman-like recursion;
• Open-loop optimisation problem: assuming to have \( \hat{x}_t \), we want to determine the optimal control input \( u_{t+1} \), i.e. it coincides with Problem (3).
Then, the robust filter solution to (12) obeys to the recursion (5) and (6) where
\[ P_{t+1} = A\Sigma T - K_t\Sigma_f T + \Sigma_f T \] (15)
and the initial conditions are \( \hat{x}_{0|0} = \hat{x}_0, P_0 = \hat{P}_0 \). Let \( e_t = x_t - \hat{x}_{t|t-1} \) denote the prediction error of \( x_t \) under the least-favourable model (solution of (12)), then \( e_t \) is Gaussian with zero mean and covariance matrix \( V_t \).

Remark 1: The right-hand side of (16) is strictly convex and monotone increasing in \( \theta \) [32]. As a consequence, the solution can be easily found by using the bisection method, see Algorithm 2.

Algorithm 2: [Bisection algorithm for computing \( \theta_t \)]
1: Input: \( \theta_t = \epsilon, \theta_t = \lambda^{-1} - \epsilon \) where \( \lambda \) max. eigenvalue of \( P_t, \epsilon > 0 \)
sufficiently small
2: Repeat
3: \( \theta_{new} = \theta_t + \epsilon \)
4: If the right hand side of (16) evaluated for \( \theta = \theta_{new} \) is smaller than \( \epsilon \)
5: then
6: \( \theta_t = \theta_{new} \)
7: else
8: \( \theta_t = \theta_{new} \)
9: endif
10: Until \( |\theta_t - \theta_{new}| \leq \epsilon \)

Proposition 1: Consider the robust filter (12) where \( (A,C) \) is reachable and \( (A,C) \) is observable. \( \lambda > 0 \) in (11) sufficiently small then \( P_0 \rightarrow P, V_0 \rightarrow V, \theta_0 \rightarrow \theta, \theta_0 \rightarrow \theta, K \rightarrow K_k \) as \( t \rightarrow \infty \), that is the filter gain converges. Moreover, \( \theta > 0, \theta > 0 \) and \( K \) is such that matrix \( -KC \) is Schur stable meaning that in the steady state the prediction error \( e_t \) is with bounded covariance matrix.

The choice of the tolerance parameter \( \epsilon \) is a delicate step. In the design phase, the user typically has: (i) an accurate and complex model of the actual process; in practice this model can be understood as an exact description of the actual process so that \( \Sigma \) is known and coincides with this model; (ii) a simple model \( \Sigma \) of the actual process of the form (1). Since \( \Sigma \) and \( \Sigma \) are known, then we can compute \( \Delta(f_t, f) \) defined in (10). Therefore, as an initial guess the parameter \( \epsilon \) can be chosen as the average of \( \Delta(f_t, f) \) over the interval of interest \( t = 1...T_{MAX} \).

The constraint \( \hat{f}_t \in \mathcal{S}_f, \epsilon = \mathcal{D}(\hat{f}_t, f), \) in (12) can be made soft by adding a penalty term in the objective function
\[ \hat{x}_{t+1|t} = \arg\max_{\hat{f}_t \in \mathcal{S}_f} E_{\mathcal{F}}(\|e_{t+1} - g(\hat{y}_t)\|^2 | Y_{t-1}) \] (16)
where
\[ E_{\mathcal{F}}(\|e_{t+1} - g(\hat{y}_t)\|^2 | Y_{t-1}) = \int_{\mathcal{F}^t} \hat{f}_t(\|e_{t+1} - g(\hat{y}_t)\|^2 | Y_{t-1}) d\mu_t \] (17)
and \( \mathcal{F} = \{ \hat{f}_t(\|e_{t+1} - g(\hat{y}_t)\|^2 | Y_{t-1}) \} \) where \( \mathcal{F} = \{ \hat{f}_t(\|e_{t+1} - g(\hat{y}_t)\|^2 | Y_{t-1}) \} \) and \( \hat{f}_t \) is fixed a priori. It is worth noting that the solution to (17) is the risk-sensitive filter with risk sensitivity parameter \( \theta_t \), see [24].

Theorem 2: Let \( u_t = a(Y_t) + b(r_t) \), where \( a(Y_t) \) is a linear function of \( Y_t \) and \( b(r_t) \) a function of a deterministic process \( r_t \). Then, the robust filter solution to (12) obeys to the recursion (5) and (6) where
\[ \theta_t > 0 \] is the unique solution to
\[ -\lambda \Delta_{t} f_t - \lambda \Delta_f f_t - I = c, \] (18)
and
\[ \lambda = \lambda(1 - \theta_t) \] is Schur stable meaning that in the steady state the prediction error \( e_t \) is with bounded covariance matrix \( V_t \).
and the initial conditions are \( x_{n-1} = \hat{x}_0 \), \( P_0 = P_0 \) positive definite. Let \( e_t = x_t - \hat{x}_{t-1} \) denote the prediction error of \( x_t \) under the least-favourable model (solution of (17)), then \( e_t \) is Gaussian with zero mean and covariance matrix \( V_t \).

**Remark 2:** The risk-sensitive filter of Theorem 2 is not well defined for any \( P_0 \) on the other hand it is possible to impose some conditions on \( P_0 \) to guarantee the evolution of \( P_t \) is on the cone of the positive definite matrices [33]. To avoid such a situation it is possible to substitute (18) with

\[
V_t = F_t \exp(\hat{\Theta} F_t^T F_t) F_t^T
\]

where \( F_t \) is the Cholesky decomposition of \( P_t \), i.e. \( P_t = F_t F_t^T \).

Relation (19) is given by solving the mini–max problem (17) by replacing \( \delta(f,f) \) with the \( r \)-divergence [34], between \( f_1 \) and \( f_t \) with parameter \( r = 1 \) [20].

**Proposition 2:** Consider the risk-sensitive filter (17) where \((A,G)\) is reachable and \((A,C)\) is observable. For \( \theta > 0 \) sufficiently small then \( P_t \rightarrow P_f \), \( V_t \rightarrow V_f \), \( L_t \rightarrow L_f \), \( K_t \rightarrow K_f \) as \( t \rightarrow \infty \). Accordingly, the filter gain converges. Moreover, \( P > 0, \ V > 0, \) and \( K \) is such that matrix \( A - KC \) is Schur stable meaning that in the steady state the prediction error \( e_t \) is bounded covariance matrix.

To recap, in the presence of an input \( u_t = a_t(Y_t) + b_t(r_t) \) we have modified the robust Kalman filter and the risk-sensitive filter by adding the term \( Bu_t \) in (6). Even if the modification is very intuitive, it is not straightforward to see that the modified filters are still robust (in the precise sense of [24] and [16], respectively). Theorems 1 and 2 showed that robustness is preserved in this case.

### 3.2 Robust MPC law

The robust MPC law that we propose is constituted by two steps: first, we compute a robust estimate of \( x_t \) given \( Y_t \) according to (12); second, the optimal input control \( u_{kt} \) is computed as in (4). Since \( u_{kt} \) is a function of \( \{r_t, \hat{x}_{kt}\} \) and thus a function of \( \{r_t, Y_t\} \), Theorem 1 holds and the robust filter (12) obeys to a Kalman-like recursion. The resulting robust MPC law is outlined in Algorithm 3.

**Algorithm 3:** [Robust MPC law]

1. Collect the new data \( y_t \)
2. Find \( \theta \) s.t. \(- \log \det(I - \theta P_t)^{-1} + \text{tr}(I - \theta P_t)^{-1} - I) = c \)
3. \( V_t = (P_t^{-1} - \theta I)^{-1} \)
4. \( L_t = V_t C^T (C V_t C^T + DD^T)^{-1} \)
5. \( \hat{x}_{kt} = \hat{x}_{kt-1} + L_t (y_t - C \hat{x}_{kt-1}) \)
6. \( u_{kt} = (L_t 0 \cdots 0 \Theta^T Q \Theta + R^{-1} \Theta Q (r_t - \Psi \hat{x}_{kt}) \)
7. Apply \( u_{kt} \) to the system
8. \( K_t = A \hat{x}_{kt} \)
9. \( P_{kt} = A V_t A^T - K_t (C V_t C^T + DD^T) K_t^T + GG^T \)
10. \( \beta_t = \beta_{kt} + K_t (y_t - C \hat{x}_{kt-1}) + G \hat{\beta}_t \)
11. \( t \leftarrow t + 1 \)
12. Go to 1

To avoid the computation of \( \beta_t \) in Step 2, we can approximate this robust filter with the risk-sensitive filter in Theorem 2. More precisely, since we know that \( \theta_t \rightarrow \hat{\theta} \) as \( t \rightarrow \infty \), we can approximate it with the risk-sensitive filter (17) with risk sensitivity parameter equal to \( \hat{\theta} \).

### 4 MPC of a servomechanism system

To test the effectiveness of the robust MPC proposed in Section 3, we consider the servomechanism system presented in [35]. It consists of a DC motor, a gear-box, an elastic shaft and a load as depicted in Fig. 1. The input is the voltage applied \( V \) and the output is the load angle \( \theta_t \). Similar mechanisms are often found in the industry for a wide variety of applications.

Since standard MPC is based on the assumption that the underlying model is linear, the general approach is to approximate the system with a linear model, possibly neglecting the non-linear dynamics. This approximation can be accurate enough as far as conventional control problems are concerned, however it may lead to intolerable errors when the DC motor operates at low speeds and rotates in two directions or when is needed a high precision control. In this situation, indeed, are significant the effects of the Coulomb and the deadzone frictions which exhibit non-linear behaviours in certain regions of operation. Next, we describe the simulation setup and we show the simulation results obtained by applying the standard MPC and the proposed robust MPC.

#### 4.1 Underlying non-linear model

We now present the complete model including the non-linear dynamics that will represent the actual model in our simulations. The equations describing the physics of the system are

\[
J_\ell \dot{\theta}_t = \rho T_s + \beta_\ell (\theta_t - \theta_{\ell 0}) - T_{f,\ell} (\dot{\theta}_t)
\]

\[
J_m \dot{\theta}_m = T_m - T_\beta (\theta_m) - T_{f,m} (\dot{\theta}_m)
\]

\[
T_m = K J_m
\]

\[
V = R I_m + L J_m + E_m
\]

\[
E_m = K \theta_m
\]

\[
T_{f,\ell} = k_0 (\theta_t - \hat{\theta}_t)
\]

where \( \theta_t \) denotes the load angle; \( \theta_m \) the motor angle; \( T_m \) the torque generated by the motor; \( E_m \) the back electromotive force; \( V \) the motor armature voltage; \( I_m \) the armature current; \( T_\ell \) the torsional torque; \( J_\ell \) the load inertia. Moreover \( T_{f,\ell}(\dot{\theta}_t) \) and \( T_{f,m}(\theta_m) \) represent the Coulomb and the deadzone frictions on the load and on the motor, as described in detail in [36]

\[
T_{f,\ell}(\dot{\theta}_t) = a_0 e^{\alpha_0 |\dot{\theta}_t|} \text{sgn}(\dot{\theta}_t) + a_1 e^{\alpha_1 |\dot{\theta}_t|} \text{sgn}(\dot{\theta}_t)\]

\[
T_{f,m}(\theta_m) = a_0 e^{\alpha_0 |\theta_m|} \text{sgn}(\theta_m) + a_1 e^{\alpha_1 |\theta_m|} \text{sgn}(\theta_m)
\]

where the function \( \text{sgn} \) is defined as
with the risk-sensitive filter in Theorem 2 (RS-MPC). The nominal values of the parameters of the servomechanism system are reported in Table 1. In practice, these values are not accurate because they are difficult to estimate. Accordingly, we introduce two kinds of possible parameter perturbations, \( \varepsilon_{\text{max}} \) and \( \varepsilon_{\text{min}} \), expressing the percentage of the relative error that each nominal value can be affected by. In the actual model every nominal parameter is perturbed in this way: if the nominal value is considered reliable enough it is perturbed by \( \pm \varepsilon_{\text{max}} = \pm 5\% \), otherwise by \( \pm \varepsilon_{\text{min}} = \pm 10\% \) (see Table 2).

### 4.2 Linear model for MPC

We now derive the nominal model. To obtain a linearised model of the servomechanism system, we eliminate the non-linear dynamics (20) and (21) and set \( L = 0 \). The dynamic equations resulting from these simplifications are

\[
\begin{align*}
J_1 \dot{\theta}_1 &= \rho T_1 - \rho_s \dot{\theta}_1 \\
J_0 \dot{\theta}_0 &= T_m - T_1 - \rho_s \dot{\theta}_0
\end{align*}
\]

wherein we consider the nominal parameters in Table 1. Defining as state vector \( x_t = [\theta_1 \; \dot{\theta}_1 \; \theta_0 \; \dot{\theta}_0 \; \theta_m \; \dot{\theta}_m] \), we obtain a continuous-time state-space linear model of type

\[
\begin{align*}
\dot{x}_t &= \tilde{A} x_t + \tilde{B} u_t + \tilde{G} w_t \\
y_t &= \tilde{C} x_t + \tilde{D} u_t
\end{align*}
\]

where \( u_t \) is the motor armature voltage, \( y_t \) the load angle, \( w_t \) is the normalised Wiener process, and matrices \( \tilde{G}, \tilde{D} \) are chosen heuristically in such a way to compensate the approximations made before. Lastly, this model was discretised with sampling time \( T = 0.1 \) s obtaining in this way a model of type (1).

### 4.3 Results

In this section, we want to show the improvement in performance brought by the MPC controller equipped with the robust Kalman filter in Theorem 1 (R-MPC). As terms of comparison we consider two other controllers: standard MPC (S-MPC) and MPC equipped with the risk-sensitive filter in Theorem 2 (RS-MPC).

More specifically, we study their performances in response with the reference trajectory set to \( r_t = \pi/2 \) rad, \( t \geq 0 \) with \( y_0 = 0 \) rad. The initial state \( x_0 \) is assumed to be zero mean and with variance equal to the variance of the state-process noise in (1). Regarding the parameters of MPC in the cost function (2) we consider the weight matrices \( Q_t = 0.1 \), \( R = 0.1 \) and \( H_p = 10 \), \( H_s = 3 \). In particular, the effects of the choice of \( H_p \) and \( H_s \) are further analysed in the last simulation (see Fig. 3).

In the first simulation, we consider the ideal situation in which the actual model coincides with the one of Section 4.2, that is the actual and the nominal models coincide. As can be seen in Fig. 4, S-MPC can track the reference trajectory likewise R-MPC (with \( c = 10^{-3} \)). However, it is worth noting that the input applied by R-MPC is less smooth (see Fig. 5).

The differences between the three controllers are clearly visible. In particular, it is evident how R-MPC is able to provide an adequate control whereas RS-MPC and S-MPC do not. To give a quantitative analysis, in the first simulation we can see how after 5 s the error on the output is comprised within 5% of the desired trajectory for all the controls (i.e. S-MPC, RS-MPC and R-MPC). On the other hand, in the second simulation the error on the output is comprised within 5% of the desired trajectory after 8 s for R-MPC, after 19 s for RS-MPC and the one of S-MPC never reaches this accuracy in 35 s. Again, R-MPC is the unique control providing similar performances both in the first and in the second simulation. On the other hand, R-MPC requires more energy (see Fig. 7 on the interval 3÷10 s) than S-MPC. Finally, RS-MPC constitutes an in-between solution in terms of performance.

### Table 1 Nominal parameters of the servomechanism system

| Symbol | Value (MKS) | Meaning            |
|--------|-------------|--------------------|
| \( L \) | 0           | armature coil inductance |
| \( J_m \) | 0.5         | motor inertia      |
| \( \rho_m \) | 0.1         | motor viscous friction coefficient |
| \( R \) | 20          | coil resistance of armature |
| \( K_t \) | 10          | motor constant      |
| \( \rho \) | 20          | gear ratio          |
| \( k_0 \) | 1280.2      | torsional rigidity  |
| \( J_f \) | 25          | load inertia        |
| \( \rho_f \) | 25          | load viscous friction coefficient |

### Table 2 Real parameters of the servomechanism system

| Symbol | Value (MKS) | Meaning            |
|--------|-------------|--------------------|
| \( L \) | 0.8         | armature coil inductance |
| \( J_m \) | 0.5(1 + \( \varepsilon_{\text{max}} \)) | motor inertia |
| \( \rho_m \) | 0.1(1 + \( \varepsilon_{\text{max}} \)) | motor viscous friction coefficient |
| \( R \) | 20(1 + \( \varepsilon_{\text{max}} \)) | resistance of armature |
| \( K_t \) | 10(1 + \( \varepsilon_{\text{max}} \)) | motor constant      |
| \( \rho \) | 20(1 + \( \varepsilon_{\text{max}} \)) | gear ratio          |
| \( k_0 \) | 1280.2(1 + \( \varepsilon_{\text{max}} \)) | torsional rigidity  |
| \( J_f \) | 25(1 - \( \varepsilon_{\text{max}} \)) | nominal load inertia |
| \( \rho_f \) | 25(1 + \( \varepsilon_{\text{max}} \)) | load viscous friction parameter |
| \( \alpha_{\ell} \) | [0.5 10 0.5] | load non-linear friction parameters |
| \( \alpha_{\theta, \ell, \alpha_{\ell}} \) | [0.1 2 0.5] | motor non-linear friction parameters |
uniform distribution probability. The unique exception is for the load inertia $J_\ell$ which is perturbed randomly with a relative error uniformly distributed inside the interval $[-80\%, 80\%]$. In this way, we consider the situation in which the servomechanism system is connected to different loads during its operational life. Then, we apply the MPC controls and for each of them we compute the MSE of the output with respect to the reference signal during the first 20 s (in which the control with R-MPC usually approaches the steady state, as seen in Fig. 6).

As expected, R-MPC performs more accurately than S-MPC and, interestingly enough, higher values of $c$ correspond to a diminished error of the MSE of R-MPC. On the other hand, higher values of $c$ correspond to high-energy control inputs.

As a last simulation, we analyse the influence of the length of the prediction and control horizons on the performance. To this aim, we have carried out the third simulation for different choices of the horizons. As representative examples, in Fig. 3 we report the results corresponding to $H_p = 10$ and $H_u = 8$ while the lower one to $H_p = 15$ and $H_u = 3$.

The first case shows that a large value for the control horizon significantly reduces the tracking error for all the MPC controllers. In particular, R-MPC1 provides the best performance while S-MPC provides the worst one. Conversely, a large value for the prediction horizon (second case) is detrimental to the performance of the MPC controllers. In particular, all the MPC controllers perform in a similar way and R-MPC3 is slightly better than the others. Finally,
we have noticed that the control action is very responsive with $H_u$ large and very slow with $H_p$ large.

5 Conclusions

MPC is nowadays one of the primary methods to address process control and is indeed employed in a wide variety of applications. One main challenge that is now presented to the research community is to find a way to gather, in the same controller, the advantages of MPC with robustness properties.

In this paper, we explored the idea to build a controller which is constituted by the usual MPC algorithm equipped with a robust Kalman filter. This appears as a very attractive option since it would meet the challenge above; in fact the design of MPC remains untouched with all of its benefits and the robust filtering can potentially compensate inaccuracies in modelling. Moreover, it would not deteriorate the computational cost from standard MPC.

Therefore we tested the controller taking into consideration a servomechanism system characterized by non-linear dynamics. In order to obtain more realistic simulations, a margin of error on the values of the system parameters was introduced. To assess the capabilities of this controller, we evaluated its efficiency with respect to classic MPC and to MPC equipped with the risk-sensitive Kalman filter.

Overall, the control system proved to be able to compensate errors in modelling significantly more effectively than the other two. In conclusion, the controller designed in this paper appears to be a viable solution to obtain all of the advantages of MPC while guaranteeing at the same time high robustness.

6 References

[1] Bagheri, P., Ali, P.B., Sedigh, A.K., et al.: ‘Analytical approach to tuning of model predictive control for first-order plus dead time models’, IET Control Theory Appl., 2013, 7, (14), pp. 1806–1817

[2] Camacho, E.F., Bordons, C.: ‘Model predictive control in the process industry’ (Springer Science & Business Media, London, United Kingdom, 2012)

[3] Mayne, D.Q.: ‘Model predictive control: recent developments and future promise’, Automatica, 2014, 50, (12), pp. 2966–2986

[4] Qin, S.J., Badgwell, T.A.: ‘A survey of industrial model predictive control technology’, Control Eng. Pract., 2003, 11, (7), pp. 733–764

[5] Shi, T., Su, H.: ‘Sampled-data MPC for LPV systems with input saturation’, IET Control Theory Appl., 2014, 8, (17), pp. 1781–1788

[6] Bemporad, A., Morari, M.: ‘Robust model predictive control: a survey’, 2007, (8), pp. 219–224

[7] Calafiore, G.C., Fagiano, L.: ‘Robust model predictive control via scenario optimization’, IEEE Trans. Autom. Control, 2013, 58, (1), pp. 219–224

[8] Cao, G., Lai, E.M.K., Alam, F.: ‘Gaussian process model predictive control of unknown nonlinear systems’, IET Control Theory Appl., 2017, 11, (5), pp. 703–713

[9] Jalali, A.A., Namid, V.: ‘A survey on robust model predictive control from 1999–2006’, Int. Conf. on Computational Intelligence for Modelling Control and Automation and Int. Conf. on Intelligent Agents, Web Technologies and Internet Commerce, Sydney, NSW, Australia, 2006

[10] Kothare, M.V., Balakrishnan, V., Morari, M.: ‘Robust constrained model predictive control using linear matrix inequalities’, Automatica, 1996, 32, (10), pp. 1361–1379

[11] Liu, Y.-H.: ‘Saturated robust adaptive control for uncertain nonlinear systems using a new approximate model’, IET Control Theory Appl., 2017, 11, (6), pp. 870–876

[12] De Nicolao, G., Magni, L., Scattolini, R.: ‘Robust predictive control of systems with uncertain impulse response’, Automatica, 1996, 32, (10), pp. 1475–1479

[13] Onuoko, P.E., Jaimoukha, I.M.: ‘Robust model predictive control based on mixed $H_2$/$H_{\infty}$ control approach’. Proc. of the European Conf., Budapest, Hungary, August 2009, pp. 2223–2228

[14] Zhang, L., Wang, Z., Ge, Y., et al.: ‘Robust distributed model predictive control for uncertain networked control systems’, IET Control Theory Appl., 2014, 8, (17), pp. 1843–1851

[15] Maciejowski, J.: ‘Discussion on “min-max model predictive control of nonlinear systems: a unifying overview on stability”’, Eur. J. Control, 2009, 15, (1), pp. 22–25

[16] Levy, B.C., Nikoukhah, R.: ‘Robust state space filtering under incremental model perturbations subject to a relative entropy tolerance’, IEEE Trans. Autom. Control, 2013, 58, (10), pp. 2611–2618

[17] Hansen, L., Sargent, T.: ‘Recursive robust estimation and control without commitment’, J. Econ. Theory, 2007, 136, (1), pp. 1–27

[18] Hansen, L., Sargent, T.: ‘Robustness’ (Princeton University Press, Princeton, NJ, 2008)

[19] Shi, T., Su, H.: ‘Sampled-data MPC for LPV systems with input saturation’, Automatica, 2014, 50, (4), pp. 892–904

[20] Zorzi, M.: ‘A new interpretation of the dual problem of the TREERE-like approaches’, Automatica, 2015, 62, pp. 87–94

[21] Banavar, R.N., Speyer, J.L.: ‘Properties of risk-sensitive filters/estimators’, IEEE Trans. Control Theory Appl., 1998, 145, (1), pp. 106–112

[22] Boel, R.K., James, M.R., Petersen, I.R.: ‘Robustness and risk-sensitive filtering’, IEEE Trans. Autom. Control, 2002, 47, (3), pp. 451–461

[23] Hassibi, B., Sayed, A., Kailath, T.: ‘Indefinite-quadratic estimation and control – a unified approach to H2 and H\infty theories’ (Society for Industrial and Applied Mathematics, Philadelphia, 1999)

[24] Yang, X., Ugrinovskii, V., Petersen, I.: ‘Robust finite time horizon minimax filtering for discrete-time stochastic uncertain systems’, Syst. Control Lett., 2004, 52, pp. 99–111

[25] Maciejowski, J.M.: ‘Predictive control: with constraints’ (Pearson Education, London, United Kingdom, 2001)

[26] Liu, S., Song, Y., Wei, G., et al.: ‘RMPC-based security problem for polytopic uncertain system subject to deception attacks and persistent disturbances’, IET Control Theory Appl., 2017, 11, (10), pp. 1611–1618

[27] Chatterjee, D., Hokayem, P., Lygeros, J.: ‘Stochastic receding horizon control with bounded control inputs: a vector space approach’, IEEE Trans. Autom. Control, 2011, 56, (11), pp. 2704–2710

[28] Zorzi, M., Li, H.: ‘Saturated robust adaptive control for uncertain non-linear systems’, IET Control Theory Appl., 2017, 11, (3), pp. 219–224

[29] Zorzi, M., Levy, B.C.: ‘On the convergence of a family of robust Kalman filters under model uncertainty’, Automatica, 2017, 83, pp. 133–140

[30] Zorzi, M., Levy, B.C.: ‘On the convergence of a risk-sensitive like filter’. 54th IEEE Conf. on Decision and Control, Osaka, Japan, December 2015

[31] Zorzi, M., Zorzi, M.: ‘A contraction analysis of the convergence of risk-sensitive filters’, SIAM J. Optim. Control, 2016, 54, (4), pp. 2154–2173

[32] Zorzi, M.: ‘Multivariate spectral estimation based on the concept of optimal prediction’, IEEE Trans. Autom. Control, 2015, 60, (6), pp. 1647–1652

[33] Bemporad, A., Mosca, E.: ‘Fulfilling hard constraints in uncertain linear systems by reference managing’, IET Control Theory Appl., 2013, 7, (14), pp. 1806–1817

[34] Bemporad, A., Morari, M.: ‘Robust model predictive control: a survey’, 2007, (8), pp. 219–224

[35] Calafiore, G.C., Fagiano, L.: ‘Robust model predictive control via scenario optimization’, IEEE Trans. Autom. Control, 2013, 58, (1), pp. 219–224

[36] Cao, G., Lai, E.M.K., Alam, F.: ‘Gaussian process model predictive control of unknown non-linear systems’, IET Control Theory Appl., 2017, 11, (5), pp. 703–713
\[
\min_{\tilde{g}} \max_{\eta \in \mathcal{D}} \int_{\mathbb{R}^{n+1}} \| x - \tilde{g}(y) \|_F^2 \tilde{j}(z) \, dz
\] (23)

where \( \mathcal{D} \) denotes the set of all estimators \( \tilde{g}(y) \) with finite second order moments with respect to any \( \tilde{f} \in \mathcal{D} \) and \( \delta = \{ \tilde{f} : \mathcal{D}(f, \delta) \leq c \} \). Then, the least favourable density \( \tilde{f} \) solution to (36) is such that
\[
\tilde{m}_c = m_c, \quad \tilde{K}_c = \begin{bmatrix} \tilde{K}_{\tilde{c}} & \tilde{K}_{\tilde{c},x} \\ \tilde{K}_{\tilde{c},x} & \tilde{K}_{\tilde{c}} \end{bmatrix}. \quad (24)
\]

Let
\[
P = K_x - K_{x,\tilde{c}} K_{\tilde{c},x}^{-1} K_x,
V = \tilde{K}_x - K_{x,\tilde{c}} K_{\tilde{c},x}^{-1} K_{x,\tilde{c}}
\]
be the nominal and the least favourable, respectively, a posteriori error covariance matrix of \( x \) given \( y \). Then \( V = (P^{-1} - \theta I)^{-1} \) and \( \theta > 0 \) is the unique solution to
\[
-\log \det (I - \theta P)^{-1} + \operatorname{tr}(I - \theta P)^{-1} - I = c. \quad (26)
\]
The robust estimator solution to (36) is the Bayes estimator
\[
g(y) = m_c + K_c^{-1}(y - m_c). \quad (27)
\]

In view of Lemma 1, it is worth noting that the estimation error \( e := x - g(y) \) under the least favourable model is Gaussian and with zero mean and covariance matrix \( V \).

We now proceed to prove the statement. Assume that the statement holds at time \( t \), that is the robust estimator of \( x_t \) given \( y_{t-1} \) is \( \hat{x}_{t-1} = E_f[x_t|y_{t-1}] \) and the prediction error \( e_t = x_t - \hat{x}_{t-1} \) is Gaussian with zero mean and covariance matrix \( V_t \). We consider now the nominal model (1). Since \( a_t(Y_t) \) is a linear function, we have
\[
u_t = \tilde{a}_t(Y_{t-1}) + J_t y_t + b_t(r_t)
= \tilde{a}_t(Y_{t-1}) + J_t C x_t + J_t D v_t + b_t(r_t)
\]
where \( \tilde{a}_t(Y_{t-1}) \) is a linear function of \( Y_{t-1} \) and \( J_t \) a matrix. Then, we have
\[
f_t(z_t|Y_{t-1}) \sim \mathcal{N}\left( \begin{bmatrix} \tilde{A}_t & \tilde{S}_t \end{bmatrix} \tilde{K}_{t-1} + \begin{bmatrix} \tilde{S}_t \end{bmatrix} \tilde{K}_c \right) \quad (29)
\]
with
\[
\tilde{K}_c = \begin{bmatrix} K_{c,\tilde{c}} & K_{c,x,\tilde{c}} \\ K_{x,\tilde{c}} & K_{\tilde{c}} \end{bmatrix}
= \begin{bmatrix} \tilde{A}_t & \tilde{C}_t \end{bmatrix} \begin{bmatrix} \tilde{G}_t \end{bmatrix} \begin{bmatrix} C_t \end{bmatrix} \quad (30)
\]
\[
\tilde{S}_t = B(\tilde{a}_t(Y_{t-1}) + b_t(r_t)), \quad \tilde{A}_t = A + BJ_t C \quad \text{and} \quad \tilde{G}_t = G + BJ_t D.
\]
Applying Lemma 1, we have that \( \tilde{f}_t(z_t|Y_{t-1}) \sim \mathcal{N}\left( \begin{bmatrix} \tilde{A}_t & \tilde{S}_t \end{bmatrix} \tilde{K}_{t-1} + \begin{bmatrix} \tilde{S}_t \end{bmatrix} \tilde{K}_c \right) \quad (31)\)
where
\[
\tilde{K}_c = \begin{bmatrix} \tilde{K}_{c,\tilde{c}} & K_{c,x,\tilde{c}} \\ K_{x,\tilde{c}} & K_{\tilde{c}} \end{bmatrix}. \quad (32)
\]

Accordingly
\[
\hat{x}_{t+1|t} = E_f[x_{t+1}|Y_{t-1}]
= \hat{A}_t \hat{x}_{t-1} + \hat{S}_t + K_{c,\tilde{c}}(y_t - C \hat{x}_{t-1}) \quad (33)
\]
and \( e_{t+1} = x_{t+1} - \hat{x}_{t+1|t} \) is Gaussian with zero mean and nominal and least favourable covariance matrix at time \( t + 1 \), respectively
\[
P_{t+1} = K_{c,\tilde{c}} + K_{c,x,\tilde{c}} K_{\tilde{c}}^{-1} K_{c,\tilde{c}} \quad (34)
\]
where \( \theta_t \) is the unique solution to (16). By (33) and (30) we have
\[
\hat{x}_{t+1|t} = \hat{A}_t \hat{x}_{t-1} + \hat{S}_t + K_{c,\tilde{c}}(y_t - C \hat{x}_{t-1})
+ (\hat{A}_t V_t \hat{A}_t^T + \hat{G}_t \hat{G}_t^T) K_{c,\tilde{c}}^{-1} (\hat{A}_t V_t \hat{A}_t^T + \hat{G}_t \hat{G}_t^T)^T
\]
which coincides with (6) where \( \hat{K}_c \) has been defined in (14). By (34) and (30) we have
\[
P_{t+1} = \hat{A}_t V_t \hat{A}_t^T + \hat{G}_t \hat{G}_t^T
+ (\hat{A}_t V_t \hat{A}_t^T + \hat{G}_t \hat{G}_t^T) K_{c,\tilde{c}}^{-1} (\hat{A}_t V_t \hat{A}_t^T + \hat{G}_t \hat{G}_t^T)^T
\]
which coincides with (15). Regarding the update (5), it can be derived with the usual argumentations.

7.2 Proof of Proposition 1
It is sufficient to note that the recursion (13)-(15) is the one studied in [37] with \( \tau = 0 \). In [37, Theorem 4.1] it was shown that such a recursion converges provided that \( \tau > 0 \) is sufficiently small. Moreover, matrix \( A - KC \) is Schur stable.

7.3 Proof of Theorem 2
To prove the statement we need the following result, see [31, Corollary 3.1] with \( \tau = 0 \).

**Lemma 2:** Let \( z = [x^T y^T] \) be a random vector with nominal and actual probability density, respectively, \( f \sim \mathcal{N}(m_0, K_0) \) and \( \bar{f} \sim \mathcal{N}(m_0, \bar{K}_0) \). We conformably partition the mean and the covariance of \( f \).
and likewise for \( \tilde{m}_z \) and \( \tilde{K}_z \). Consider the min problem

\[
\min_{g \in \mathcal{G}} \int_{\mathbb{R}^n} \| x - g(y) \|^2 f(z) dz - \theta^{-1} \mathcal{D}(\tilde{f}, f)
\]

where \( \mathcal{G} \) denotes the set of all estimators \( g(y) \) with finite second order moments with respect to any \( \tilde{f} \in \mathcal{D} \) and \( \mathcal{D} = \{ \tilde{f} : \mathcal{D}(\tilde{f}, f) \leq \infty \} \). Then, the least favourable density \( \tilde{f} \) solution to (36) is such that

\[
\tilde{m}_z = m_z, \quad \tilde{K}_z = \begin{bmatrix} K_x & K_{x,y} \\ K_{y,x} & K_y \end{bmatrix}
\]

(37)

Let

\[
m_x = \begin{bmatrix} m_x \\ m_y \end{bmatrix}, \quad K_x = \begin{bmatrix} K_x & K_{x,y} \\ K_{y,x} & K_y \end{bmatrix}
\]

(35)

be the nominal and the least favourable, respectively, a posteriori error covariance matrix of \( x \) given \( y \). Then \( V = (P^{-1} - \theta I)^{-1} \). The robust estimator solution to (36) is the Bayes estimator

\[
g(y) = m_x + K_x, K_{y,x}^{-1}(y - m_y).
\]

(39)

Using the above result, the proof of the Theorem is similar to the one of Theorem 1.

7.4 Proof of Proposition 2

It is sufficient to note that the recursion (18) is the one studied in [33]. In [33, Theorem 5.3] it was shown that such a recursion converges provided that \( \theta > 0 \) is sufficiently small. Moreover, matrix \( A - \tilde{K}C \) is Schur stable.