CHALMERS

UNIVERSITY OF TECHNOLOGY

SSY281 - MODEL PREDICTIVE CONTROL

NIKOLCE MURGOVSKI

Division of Systems and Control
Department of Electrical Engineering
Chalmers University of Technology
Gothenburg, Sweden

2021-02-26

Lecture 12: Explicit control laws for constrained, linear systems

Goals for today:

- To understand how the concept of parametric programming can be applied to RHC
- To understand the principles behind so called explicit MPC

Lecture 12: Explicit control laws for constrained, linear systems

Goals for today:

- To understand how the concept of parametric programming can be applied to RHC
- To understand the principles behind so called explicit MPC

Learning objectives:

- Describe and construct MPC controllers based on a linear model, quadratic costs and linear constraints
- Describe basic properties of MPC controllers and analyse algorithmic details on very simple examples

Parametric programming

So called *parametric programming* deals with optimisation problems taking the form

$$V^*(x) = \min_u \{V(x,u) \mid u \in \mathcal{U}(x)\}, \tag{98}$$

where x is the parameter of the problem. Whereas the solution to a conventional optimisation problem is a *point* (or possibly a set), the solution to this parametric programming problem is actually a *function* $u^*(x)$ (which in the general case could be set valued).

Parametric programming

So called *parametric programming* deals with optimisation problems taking the form

$$V^*(x) = \min_u \{V(x,u) \mid u \in \mathcal{U}(x)\}, \tag{98}$$

where x is the parameter of the problem. Whereas the solution to a conventional optimisation problem is a *point* (or possibly a set), the solution to this parametric programming problem is actually a *function* $u^*(x)$ (which in the general case could be set valued).
The parametric constraint $u \in \mathcal{U}(x)$ can also be expressed as $(x,u) \in \mathcal{Z}$, where $\mathcal{Z}$ is a subset of (x,u)-space,

$$\mathcal{U}(x) = \{u \mid (x,u) \in \mathcal{Z}\}. \tag{99}$$

Parametric programming

So called *parametric programming* deals with optimisation problems taking the form

$$V^*(x) = \min_u \{V(x,u) \mid u \in \mathcal{U}(x)\}, \tag{98}$$

where x is the parameter of the problem. Whereas the solution to a conventional optimisation problem is a *point* (or possibly a set), the solution to this parametric programming problem is actually a *function* $u^*(x)$ (which in the general case could be set valued).
The parametric constraint $u \in \mathcal{U}(x)$ can also be expressed as $(x,u) \in \mathcal{Z}$, where $\mathcal{Z}$ is a subset of (x,u)-space,

$$\mathcal{U}(x) = \{u \mid (x,u) \in \mathcal{Z}\}. \tag{99}$$

The domain of the function $V^*(x)$ in (98) is the set $\mathcal{X}$, defined by

$$\mathcal{X} = \{x \mid \exists u \text{ such that } (x,u) \in \mathcal{Z}\} = \{x \mid \mathcal{U}(x) \neq \emptyset\}. \tag{100}$$

Example: Parametric quadratic programming

Consider the parametric quadratic program $\min_u \{V(x,u) \mid (x,u) \in \mathcal{Z}\}$ with

$$V(x,u) = \frac{1}{2}\left((x-u)^2 + u^2\right)$$
$$\mathcal{Z} = \{(x,u) \mid u \geq 1,\ u + x/2 \geq 2,\ u + x \geq 2\}.$$

Example: Parametric quadratic programming

Consider the parametric quadratic program $\min_u \{V(x,u) \mid (x,u) \in \mathcal{Z}\}$ with

$$V(x,u) = \frac{1}{2}\left((x-u)^2 + u^2\right)$$
$$\mathcal{Z} = \{(x,u) \mid u \geq 1,\ u + x/2 \geq 2,\ u + x \geq 2\}.$$

From $\nabla_u V(x,u) = -x + 2u$, it is clear that the unconstrained minimum is given by

$$u^*_{uc} = x/2,$$

but this solution does not fulfil the constraints for $x < 2$. Since $\nabla_u V(x,u) > 0$ for all $u > u^*_{uc}$, the constrained optimal solution $u^*(x)$ will lie on the boundary of $\mathcal{Z}$.

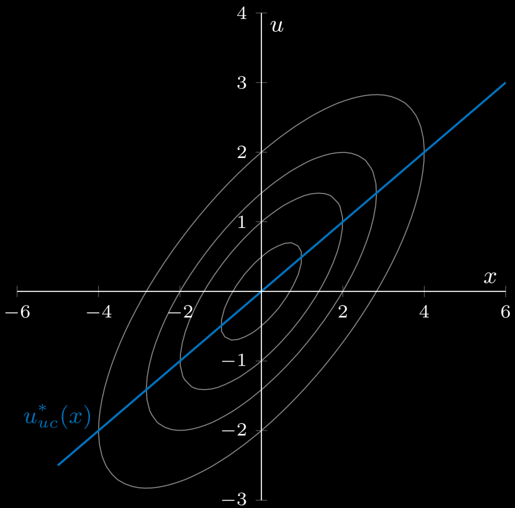


Figure 42: An illustration of the unconstrained, u^*_{uc}, and the constrained parametric solution, $u^*(x)$, to the quadratic program.

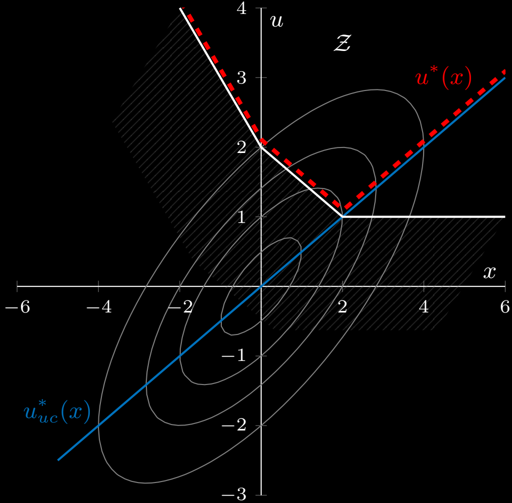


Figure 42: An illustration of the unconstrained, u^*_{uc}, and the constrained parametric solution, $u^*(x)$, to the quadratic program.

Constrained LQ control

Recall that the receding horizon controller is based on the finite-time optimisation problem

$$V_N^*(x) = \min_{u(0:N-1)} \{V_N(x, u(0:N-1)) \mid u(0:N-1) \in \mathcal{U}_N(x)\}, \tag{101}$$

where the objective is given by

$$V_N(x, u(0:N-1)) = x^\top(N) P_f x(N) + \sum_{i=0}^{N-1} \left(x^\top(i) Q x(i) + u^\top(i) R u(i) \right), \quad x(0) = x. \tag{102}$$

Constrained LQ control

Recall that the receding horizon controller is based on the finite-time optimisation problem

$$V_N^*(x) = \min_{u(0:N-1)} \{V_N(x, u(0:N-1)) \mid u(0:N-1) \in \mathcal{U}_N(x)\}, \tag{101}$$

where the objective is given by

$$V_N(x, u(0:N-1)) = x^\top(N)P_f x(N) + \sum_{i=0}^{N-1} \left(x^\top(i)Qx(i) + u^\top(i)Ru(i)\right), \quad x(0) = x. \tag{102}$$

In this formulation, the optimisation variables are the sequence of controls $u(0:N-1)$ or, equivalently, the vector $\boldsymbol{u} = \text{vec}\,(u(0), \ldots, u(N-1))$ as in (20).

Constrained LQ control

Recall that the receding horizon controller is based on the finite-time optimisation problem

$$V_N^*(x) = \min_{u(0:N-1)} \{V_N(x, u(0:N-1)) \mid u(0:N-1) \in \mathcal{U}_N(x)\}, \tag{101}$$

where the objective is given by

$$V_N(x, u(0:N-1)) = x^\top(N) P_f x(N) + \sum_{i=0}^{N-1} \left(x^\top(i) Q x(i) + u^\top(i) R u(i) \right), \quad x(0) = x. \tag{102}$$

In this formulation, the optimisation variables are the sequence of controls $u(0:N-1)$ or, equivalently, the vector $\boldsymbol{u} = \text{vec}\,(u(0), \ldots, u(N-1))$ as in (20). The states $x(i)$ in (102) can be thought of as short-hand notation for the expressions

$$x(i) = A^i x + \Gamma_i \boldsymbol{u}, \tag{103}$$

where Γ_i is the i^{th} row of the matrix Γ, see (19).

With the formulation above, all the constraints are encoded in the form of the set of allowed control sequences $u(0\!:\!N-1) \in \mathcal{U}_N(x)$, which depends on the initial state x.

With the formulation above, all the constraints are encoded in the form of the set of allowed control sequences $u(0:N-1) \in \mathcal{U}_N(x)$, which depends on the initial state x.
Taking a closer look at this set, we see that it is actually determined by the following constraints:

1. the control constraint set $\mathbb{U}$;
2. the state constraint set $\mathbb{X}$;
3. the terminal state constraint set $\mathbb{X}_f$.

With the formulation above, all the constraints are encoded in the form of the set of allowed control sequences $u(0\!:\!N-1) \in \mathcal{U}_N(x)$, which depends on the initial state x.
Taking a closer look at this set, we see that it is actually determined by the following constraints:

1. the control constraint set $\mathbb{U}$;
2. the state constraint set $\mathbb{X}$;
3. the terminal state constraint set $\mathbb{X}_f$.

Condition (1) implies that $\boldsymbol{u}$ is constrained by a polyhedral set. Conditions (2) and (3) constitute polyhedral constraints on $\{x(i)\}$, which can be translated to affine constraints expressed in terms of $(x, \boldsymbol{u})$ by using the state expressions (103).

With the formulation above, all the constraints are encoded in the form of the set of allowed control sequences $u(0\!:\!N-1) \in \mathcal{U}_N(x)$, which depends on the initial state x.
Taking a closer look at this set, we see that it is actually determined by the following constraints:

1. the control constraint set $\mathbb{U}$;
2. the state constraint set $\mathbb{X}$;
3. the terminal state constraint set $\mathbb{X}_f$.

Condition (1) implies that $\boldsymbol{u}$ is constrained by a polyhedral set. Conditions (2) and (3) constitute polyhedral constraints on $\{x(i)\}$, which can be translated to affine constraints expressed in terms of $(x, \boldsymbol{u})$ by using the state expressions (103). The conclusion is that the *implicit* constraint set $\mathcal{U}_N$ can be expressed as an *explicit*, polyhedral set

$$\mathbb{Z} = \{(x, \boldsymbol{u}) \mid F\boldsymbol{u} \leq Gx + h\} \tag{104}$$

for some matrices F, G and vector h.

Hence, the MPC optimisation problem can now be stated as

$$\begin{aligned} \min_{\boldsymbol{u}} \quad & V_N(x, \boldsymbol{u}) = \frac{1}{2}(\boldsymbol{u}^\top \tilde{R}\boldsymbol{u} + x^\top \tilde{Q}x) + \boldsymbol{u}^\top S x \\ \text{subject to} \quad & F\boldsymbol{u} \le Gx + h \end{aligned} \tag{105}$$

for suitable definitions of the matrices $\tilde{R}, \tilde{Q}$, and S, cf. (21). In the sequel, it will be assumed that the matrix

$$\mathcal{Q} = \begin{bmatrix} \tilde{Q} & S^\top \\ S & \tilde{R} \end{bmatrix}$$

is positive definite, implying that both $\tilde{R}$ and $\tilde{Q}$ are positive definite.

Solution of the parametric program

Let's first note that the parametric optimisation problem (105) can also be stated as

$$V^*(x) = \min_{\boldsymbol{u}} \{V(x, \boldsymbol{u}) \mid \boldsymbol{u} \in \mathcal{U}(x)\} \tag{106}$$

where

$$\mathcal{U}(x) = \{\boldsymbol{u} \mid (x, \boldsymbol{u}) \in \mathbb{Z}\} = \{\boldsymbol{u} \mid F\boldsymbol{u} \leq Gx + h\} \tag{107}$$

$$\mathbb{Z} = \{(x, \boldsymbol{u}) \mid F\boldsymbol{u} \leq Gx + h\} \tag{108}$$

and we have dropped the subscript N.

Solution of the parametric program

Let's first note that the parametric optimisation problem (105) can also be stated as

$$V^*(x) = \min_{\boldsymbol{u}} \{V(x, \boldsymbol{u}) \mid \boldsymbol{u} \in \mathcal{U}(x)\} \tag{106}$$

where

$$\mathcal{U}(x) = \{\boldsymbol{u} \mid (x, \boldsymbol{u}) \in \mathbb{Z}\} = \{\boldsymbol{u} \mid F\boldsymbol{u} \leq Gx + h\} \tag{107}$$

$$\mathbb{Z} = \{(x, \boldsymbol{u}) \mid F\boldsymbol{u} \leq Gx + h\} \tag{108}$$

and we have dropped the subscript N. The domain of V^* is

$$\mathcal{X} = \{x \mid \exists \boldsymbol{u} \text{ such that } (x, \boldsymbol{u}) \in \mathbb{Z}\} = \{x \mid \mathcal{U} \neq \emptyset\}.$$

Solution of the parametric program

Let's first note that the parametric optimisation problem (105) can also be stated as

$$V^*(x) = \min_{\boldsymbol{u}}\{V(x,\boldsymbol{u}) \mid \boldsymbol{u} \in \mathcal{U}(x)\} \tag{106}$$

where

$$\mathcal{U}(x) = \{\boldsymbol{u} \mid (x,\boldsymbol{u}) \in \mathbb{Z}\} = \{\boldsymbol{u} \mid F\boldsymbol{u} \leq Gx + h\} \tag{107}$$

$$\mathbb{Z} = \{(x,\boldsymbol{u}) \mid F\boldsymbol{u} \leq Gx + h\} \tag{108}$$

and we have dropped the subscript N. The domain of V^* is

$$\mathcal{X} = \{x \mid \exists \boldsymbol{u} \text{ such that } (x,\boldsymbol{u}) \in \mathbb{Z}\} = \{x \mid \mathcal{U} \neq \emptyset\}.$$

Since $\tilde{R}$ has been assumed positive definite, the solution $\boldsymbol{u}^*(x)$ is unique and its domain is $\mathcal{X}$.

Main ideas to exploit the structure of the problem

1. Pick an arbitrary state vector x, for which the optimal solution is $\boldsymbol{u}^*(x)$. Find a representation of this solution by using the fact that it is also the optimal solution of an equality constrained problem (with the subset of active inequality constraints being included as equality constraints and the remaining ones discarded).

Main ideas to exploit the structure of the problem

1. Pick an arbitrary state vector x, for which the optimal solution is $\boldsymbol{u}^*(x)$. Find a representation of this solution by using the fact that it is also the optimal solution of an equality constrained problem (with the subset of active inequality constraints being included as equality constraints and the remaining ones discarded).
2. Show that the same equality constrained problem, and its solution, is also valid for initial states close to x. In fact, there exists a polyhedron R_x^* in $\mathbb{R}^n$, containing x, such that, for each $w \in R_x^*$, $\boldsymbol{u}^*(w)$ is the solution of the optimisation problem *with the same set of equality constraints*. On the set R_x^*, $\boldsymbol{u}^*(w)$ is affine in w and $V^*(w)$ is quadratic in w.

Main ideas to exploit the structure of the problem

1. Pick an arbitrary state vector x, for which the optimal solution is $\boldsymbol{u}^*(x)$. Find a representation of this solution by using the fact that it is also the optimal solution of an equality constrained problem (with the subset of active inequality constraints being included as equality constraints and the remaining ones discarded).
2. Show that the same equality constrained problem, and its solution, is also valid for initial states close to x. In fact, there exists a polyhedron R_x^* in $\mathbb{R}^n$, containing x, such that, for each $w \in R_x^*$, $\boldsymbol{u}^*(w)$ is the solution of the optimisation problem *with the same set of equality constraints*. On the set R_x^*, $\boldsymbol{u}^*(w)$ is affine in w and $V^*(w)$ is quadratic in w.
3. Show that there are finitely many polyhedral regions $\{R_x^*\}$ covering the set of feasible states $\mathcal{X}$.

Step 1

The conditions of optimality are that $\boldsymbol{u}$ is feasible and that

$$-\nabla_{\boldsymbol{u}} V(x) = -(\tilde{R}\boldsymbol{u} + Sx) = \sum_{i\in\mathbb{A}} \mu_i F_i^\top, \quad \text{for some } \{\mu_i\} \text{ with } \mu_i \geq 0 \tag{109}$$

where F_i is the i^{th} row of F and $\mathbb{A}$ is the active set of constraints.

Step 1

The conditions of optimality are that $\boldsymbol{u}$ is feasible and that

$$-\nabla_{\boldsymbol{u}} V(x) = -(\tilde{R}\boldsymbol{u} + Sx) = \sum_{i \in \mathbb{A}} \mu_i F_i^\top, \quad \text{for some } \{\mu_i\} \text{ with } \mu_i \geq 0 \tag{109}$$

where F_i is the i^{th} row of F and $\mathbb{A}$ is the active set of constraints. The latter condition can equivalently be expressed as a set of linear inequalities (see [1]), so that the optimality conditions can be written as a set of linear inequalities:

$$\begin{aligned} F\boldsymbol{u} &\leq Gx + h \\ -L_x^*(\tilde{R}\boldsymbol{u} + Sx) &\leq 0 \end{aligned} \tag{110}$$

where L_x^* is a matrix defined by the active constraints corresponding to $\boldsymbol{u}^*(x)$.

Step 1

The conditions of optimality are that $\boldsymbol{u}$ is feasible and that

$$-\nabla_{\boldsymbol{u}} V(x) = -(\tilde{R}\boldsymbol{u} + Sx) = \sum_{i\in\mathbb{A}} \mu_i F_i^\top, \quad \text{for some } \{\mu_i\} \text{ with } \mu_i \geq 0 \tag{109}$$

where F_i is the i^{th} row of F and $\mathbb{A}$ is the active set of constraints. The latter condition can equivalently be expressed as a set of linear inequalities (see [1]), so that the optimality conditions can be written as a set of linear inequalities:

$$\begin{aligned} F\boldsymbol{u} &\leq Gx + h \\ -L_x^*(\tilde{R}\boldsymbol{u} + Sx) &\leq 0 \end{aligned} \tag{110}$$

where L_x^* is a matrix defined by the active constraints corresponding to $\boldsymbol{u}^*(x)$.
The solution $\boldsymbol{u}^*(x)$ can also be found as the solution to an equality constrained problem with only the active constraints. Denoting the subset of active constraints corresponding to x by $F_x^*\boldsymbol{u} = G_x^* x + h_x^*$, we thus have

$$\boldsymbol{u}^*(x) = \arg\min_{\boldsymbol{u}} \{V(x, \boldsymbol{u}) \mid F_x^*\boldsymbol{u} = G_x^* x + h_x^*\}.$$

Step 2

At least for points w close to x, it sounds plausible that the optimal solution $\boldsymbol{u}^*(w)$ would have the same set of active constraints as x. If so, then $\boldsymbol{u}^*(w)$ would actually be the solution of an equality constrained problem with *the same set of equality constraints*. We have not yet shown that this is the case, but let's define these solutions anyhow,

$$\boldsymbol{u}_x^*(w) = \arg\min_{\boldsymbol{u}} \{V(w, \boldsymbol{u}) \mid F_x^* \boldsymbol{u} = G_x^* w + h_x^*\}. \tag{111}$$

Here, the subscript x indicates that the solution is obtained by using the active constraints corresponding to the state x.

Step 2

At least for points w close to x, it sounds plausible that the optimal solution $\boldsymbol{u}^*(w)$ would have the same set of active constraints as x. If so, then $\boldsymbol{u}^*(w)$ would actually be the solution of an equality constrained problem with *the same set of equality constraints*. We have not yet shown that this is the case, but let's define these solutions anyhow,

$$\boldsymbol{u}_x^*(w) = \arg\min_{\boldsymbol{u}} \{V(w, \boldsymbol{u}) \mid F_x^* \boldsymbol{u} = G_x^* w + h_x^*\}. \tag{111}$$

Here, the subscript x indicates that the solution is obtained by using the active constraints corresponding to the state x.

Now, the solution of a QP with linear constraints can be computed explicitly, see (97). The result is that $\boldsymbol{u}_x^*(w)$ is affine in w and that $V_x^*(w)$ is quadratic in w:

$$\boldsymbol{u}_x^*(w) = K_x w + k_x \tag{112}$$

$$V_x^*(w) = \frac{1}{2} w^\top Q_x w + r_x^\top w + s_x. \tag{113}$$

By inserting the expression for $\boldsymbol{u}_x^*(w)$ in equation (110) and replacing x with w, we obtain a set of inequalities for w:

$$
\begin{aligned}
F(K_x w + k_x) &\leq Gw + h \\
-L_x^*(\tilde{R}(K_x w + k_x) + Sw) &\leq 0.
\end{aligned}
\tag{114}
$$

By inserting the expression for $\boldsymbol{u}_x^*(w)$ in equation (110) and replacing x with w, we obtain a set of inequalities for w:

$$\begin{aligned} F(K_x w + k_x) &\leq Gw + h \\ -L_x^*(\tilde{R}(K_x w + k_x) + Sw) &\leq 0. \end{aligned} \tag{114}$$

This set of inequalities defines a polyhedron R_x^* for w. The conclusion is that for any $w \in R_x^*$, the previously computed solution $\boldsymbol{u}_x^*(w)$ satisfies the optimality conditions for the original problem. Hence,

$$\boldsymbol{u}^*(w) = \boldsymbol{u}_x^*(w) = K_x w + k_x, \quad \forall w \in R_x^*. \tag{115}$$

By inserting the expression for $\boldsymbol{u}_x^*(w)$ in equation (110) and replacing x with w, we obtain a set of inequalities for w:

$$\begin{aligned} F(K_x w + k_x) &\leq Gw + h \\ -L_x^*(\tilde{R}(K_x w + k_x) + Sw) &\leq 0. \end{aligned} \tag{114}$$

This set of inequalities defines a polyhedron R_x^* for w. The conclusion is that for any $w \in R_x^*$, the previously computed solution $\boldsymbol{u}_x^*(w)$ satisfies the optimality conditions for the original problem. Hence,

$$\boldsymbol{u}^*(w) = \boldsymbol{u}_x^*(w) = K_x w + k_x, \quad \forall w \in R_x^*. \tag{115}$$

It follows that the value function is quadratic in R_x^*.

Step 3

- The set R_x^* is characterized by the set of active constraints, and there is a finite set of subsets of active constraints, telling us that there are finitely many polyhedral regions R_x^*.

Step 3

- The set R_x^* is characterized by the set of active constraints, and there is a finite set of subsets of active constraints, telling us that there are finitely many polyhedral regions R_x^*.
- Any arbitrary feasible $x \in \mathcal{X}$ belongs to a region R_x^*.

Step 3

- The set R_x^* is characterized by the set of active constraints, and there is a finite set of subsets of active constraints, telling us that there are finitely many polyhedral regions R_x^*.
- Any arbitrary feasible $x \in \mathcal{X}$ belongs to a region R_x^*.
- We can conclude that the feasible set $\mathcal{X}$ is partitioned by a collection of non-overlapping polyhedra $\cup\{R_x^*\} \mid x \in \mathcal{X}$.

Step 3

- The set R_x^* is characterized by the set of active constraints, and there is a finite set of subsets of active constraints, telling us that there are finitely many polyhedral regions R_x^*.
- Any arbitrary feasible $x \in \mathcal{X}$ belongs to a region R_x^*.
- We can conclude that the feasible set $\mathcal{X}$ is partitioned by a collection of non-overlapping polyhedra $\cup\{R_x^*\} \mid x \in \mathcal{X}$.
- Corresponding to this partition, the optimal solution to the parametric program (105) is piecewise affine with a value function that is piecewise quadratic.

Step 3

- The set R_x^* is characterized by the set of active constraints, and there is a finite set of subsets of active constraints, telling us that there are finitely many polyhedral regions R_x^*.
- Any arbitrary feasible $x \in \mathcal{X}$ belongs to a region R_x^*.
- We can conclude that the feasible set $\mathcal{X}$ is partitioned by a collection of non-overlapping polyhedra $\cup\{R_x^*\} \mid x \in \mathcal{X}$.
- Corresponding to this partition, the optimal solution to the parametric program (105) is piecewise affine with a value function that is piecewise quadratic.
- Moreover, as shown in [1], the value function $V^*(x)$ and the minimizer $\boldsymbol{u}^*(x)$ are continuous in $x \in \mathcal{X}$.

Example: LQ MPC for an integrator, cont'd

$$x^+ = x + u$$

with a control constraint

$$u \in \mathbb{U} = [-1, 1]$$

and with $Q = R = P_f = 1$, $N = 2$.

Example: LQ MPC for an integrator, cont'd

$$x^+ = x + u$$

with a control constraint

$$u \in \mathbb{U} = [-1, 1]$$

and with $Q = R = P_f = 1$, $N = 2$. For this example, we concluded that the receding horizon control law could be given in closed-form as

$$u(x) = \begin{cases} 1 & x \leq -5/3 \\ -3/5 \cdot x & -5/3 \leq x \leq 5/3 \\ -1 & x \geq 5/3. \end{cases}$$

Example: LQ MPC for an integrator, cont'd

$$x^+ = x + u$$

with a control constraint

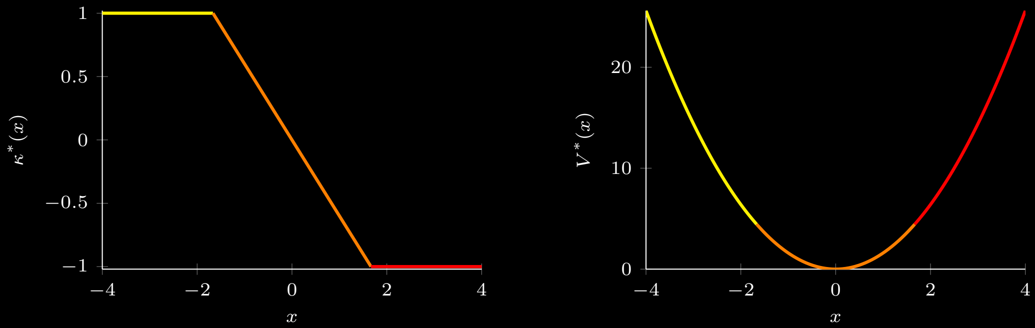
$$u \in \mathbb{U} = [-1, 1]$$

and with $Q = R = P_f = 1$, $N = 2$. For this example, we concluded that the receding horizon control law could be given in closed-form as

$$u(x) = \begin{cases} 1 & x \leq -5/3 \\ -3/5 \cdot x & -5/3 \leq x \leq 5/3 \\ -1 & x \geq 5/3. \end{cases}$$

We can thus verify that the control law is piecewise affine, as predicted, and that the partition of the state space is given by three intervals, two of which are semi-infinite.

LQ MPC for an integrator, cont'd



(a) Optimal feedback policy.

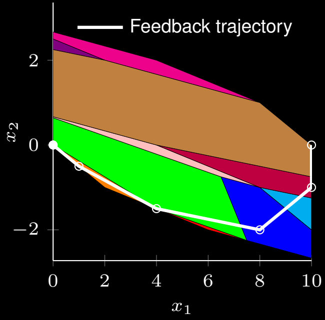
(b) Optimal value function.

Figure 43: Optimal explicit control law and value function. The colours indicate different control partitions.

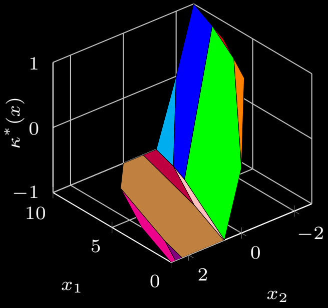
Example: High inertia system, cont'd

$$x(k+1) = \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix} x(k) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(k), \quad y(k) = \begin{bmatrix} 1 & 0 \end{bmatrix} x(k), \quad x(0) = \begin{bmatrix} 10 \\ 0 \end{bmatrix}, \quad Q = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix},$$

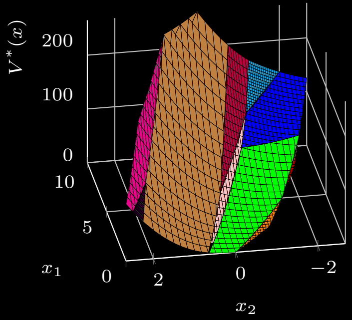
$$R = 1, \quad P_f = P_\infty, \quad x_1(k) \in [0, 10], \quad u(k) \in [-1, 1], \forall k, \quad N = 2, \quad \mathbb{X}_f = \mathcal{C}_\infty.$$



(a) Control partitions.



(b) Optimal feedback policy.



(c) Optimal value function.

Figure 44: Optimal explicit control law. The colours indicate different control partitions.

Explicit MPC strategy

1. Find a partition $\{R_{x_i}\}$ of $\mathcal{X}$.

Explicit MPC strategy

1. Find a partition $\{R_{x_i}\}$ of $\mathcal{X}$.
2. Calculate the optimal, affine control law on each of the regions R_{x_i} and store the result, i.e. controller parameters. This and the previous step can be done *off-line*.

Explicit MPC strategy

1. Find a partition $\{R_{x_i}\}$ of $\mathcal{X}$.
2. Calculate the optimal, affine control law on each of the regions R_{x_i} and store the result, i.e. controller parameters. This and the previous step can be done *off-line*.
3. During *on-line* operation, run the controller by

Explicit MPC strategy

1. Find a partition $\{R_{x_i}\}$ of $\mathcal{X}$.
2. Calculate the optimal, affine control law on each of the regions R_{x_i} and store the result, i.e. controller parameters. This and the previous step can be done *off-line*.
3. During *on-line* operation, run the controller by
   - for the measured state x, identify the region to which x belongs;

Explicit MPC strategy

1. Find a partition $\{R_{x_i}\}$ of $\mathcal{X}$.
2. Calculate the optimal, affine control law on each of the regions R_{x_i} and store the result, i.e. controller parameters. This and the previous step can be done *off-line*.
3. During *on-line* operation, run the controller by
   - for the measured state x, identify the region to which x belongs;
   - look up the controller parameters for the found region;

Explicit MPC strategy

1. Find a partition $\{R_{x_i}\}$ of $\mathcal{X}$.
2. Calculate the optimal, affine control law on each of the regions R_{x_i} and store the result, i.e. controller parameters. This and the previous step can be done *off-line*.
3. During *on-line* operation, run the controller by
   - for the measured state x, identify the region to which x belongs;
   - look up the controller parameters for the found region;
   - compute the next control signal.

Explicit MPC example

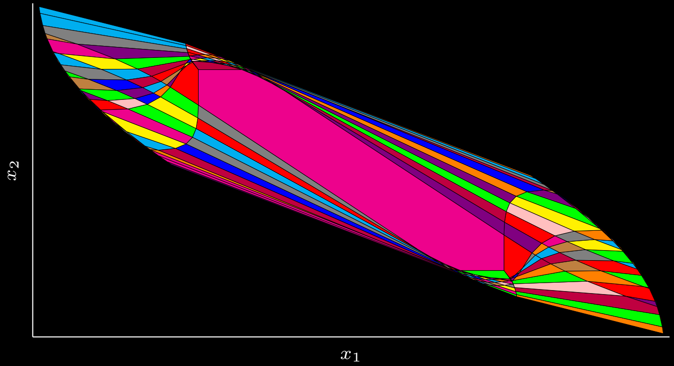


Figure 45: An illustration of a partitioned state-space for a system with 2 states and prediction horizon of 10 samples. There are 267 control partitions with a possibly different affine control law.

References

[1] J.B. Rawlings, D.Q. Mayne, and M.M. Diehl. *Model Predictive Control: Theory, Computation, and Design, 2nd edition*. Nob Hill Publishing 2017.
Available online at `https://sites.engineering.ucsb.edu/~jbraw/mpc`

[2] G. Goodwin, M.M. Seron, and J.A. De Don. *Constrained Control and Estimation*. Springer 2004.
Available online via Chalmers Library.

[3] F Borrelli, A. Bemporad, and M. Morari. *Predictive Control for Linear and Hybrid Systems*.
Available online at http://www.mpc.berkeley.edu/mpc-course-material

[4] J. Maciejowski. *Predictive Control with Constraints*. Prentice Hall 2002.

[5] S. Boyd and L. Vandenberghe. *Convex optimisation*. Cambridge University Press 2004.

[6] Diehl, M. *Real-Time Optimization for Large Scale Nonlinear Processes*. PhD thesis, University of Heidelberg, 2001.

AVANCEZ
1829

CHALMERS
UNIVERSITY OF TECHNOLOGY