Explicit MPC

Based on **Explicit Model-Predictive Control** by **Alberto Bemporad**

Hakan Asik

Outline

- Explicit & Implicit definition
- Implicit MPC vs Explicit MPC
- Explicit MPC
  - Advantages
  - Multi-parametric programming
  - Piecewise Affine functions
  - Methodology
  - Limitations
  - Applications
- Case Study

MPC

- **Definition:** An MPC uses linear plant, disturbance, and noise models to estimate the controller state and predict future plant outputs.
- **Objective:** Using the predicted plant outputs, the controller solves a quadratic programming optimization problem to determine optimal manipulated variable adjustments.

  which has 2 approaches:

  - Implicit MPC
  - Explicit MPC

Explicit & Implicit definition

- **Implicit MPC**: Traditional approach, in which MPC requires the solution of a quadratic program (QP) online to compute the control action, often restricting its applicability to slow processes.
- **Explicit MPC:** Explicit MPC completely removes the need for on-line solvers by precomputing the control law off-line, so that online operations reduce to a simple function evaluation.

Implicit MPC vs Explicit MPC

- Explicit & Implicit definition
- **Implicit MPC vs Explicit MPC**
- Explicit MPC
  - Advantages
  - Multi-parametric programming
  - Piecewise Affine functions
  - Methodology
  - Limitations
  - Applications
- Case Study

Implicit MPC vs Explicit MPC

- The differences between two can be summarized as following:

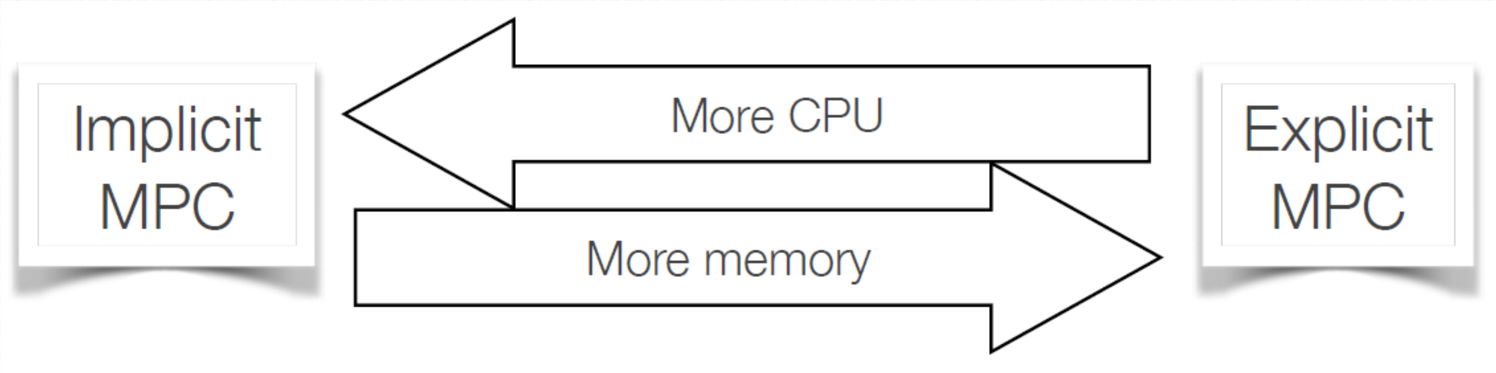
	Implicit MPC	Explicit MPC
Problem size	Large	Small
KKT solver	Online matrix inverse	Off-line factorization
Active set update	Heuristics	Sequential
Memory storage	Small	Large

	Vars	Constr.	States
Small	≤ 6	15	8
Large	≈ 500	3000	2000

Implicit MPC vs Explicit MPC

In Implicit MPC, there involves online solvers. Hence, requires larger processors.

Whereas, Explicit MPC need not process online but store a number of off-line algorithms. Thus, greater storage capacity is required



Implicit MPC vs Explicit MPC

	Implicit MPC (active-set method)	Explicit MPC (incremental regionless)
Memory storage	*H, F, G, w, E*	*G, w, E* primal optimizer, dual optimizer
KKT solver	Matrix inversions	Matrix/vector multiplication
Active set update	Identical algorithm	

Explicit MPC Simulink Block

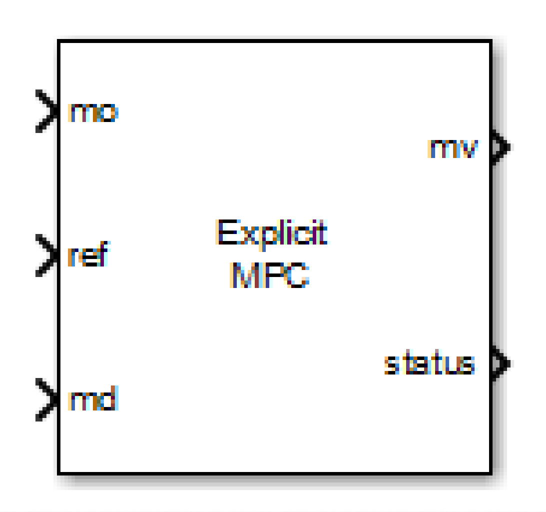
The Explicit MPC Controller block uses the following input signals:

- Measured plant outputs (mo)
- Reference or setpoint (ref)
- Measured plant disturbance (md), if any

Measured plant outputs are obtained from Implicit approach.

Hence, to create an **Explicit MPC controller**:

- first **Traditional (Implicit) MPC controller** is designed.
- then an **Explicit MPC Controller** is generated based on this traditional controller design.



Agenda

- Explicit & Implicit definition
- Implicit MPC vs Explicit MPC
- Explicit MPC
  - **Advantages**
  - Multi-parametric programming (Linear and Quadratic)
  - Piecewise Affine functions
  - Methodology (MPLP and MPQP
  - Limitations
  - Applications
- Case Study

Advantages of Explicit MPC

- It can be noted that Explicit MPC:
  - Does off-line and simpler computations.
  - Requires fewer run-time computations, thus useful for system with low sample rates.
  - Requires processors of less computational capacity.

MPC Problem

- In practical, an MPC should have a QP solution method:
  - embedded in the control hardware
  - the method must be fast enough to provide a solution within short sampling intervals
  - require simple hardware, limited memory to store the data defining the optimization problem and the code implementing the algorithm itself
  - a simple program code
  - and a good **worst-case** estimates of the execution time to meet real-time system requirements

Agenda

- Explicit & Implicit definition
- Implicit MPC vs Explicit MPC
- Explicit MPC
  - Advantages
  - **Multi-parametric programming**
  - Piecewise Affine functions
  - Methodology
  - Limitations
  - Applications
- Case Study

Multi-parametric programming

- MPC takes a different approach to meet the above requirements, where Multi-Parametric Quadratic Programming is proposed to pre-solve the QP off-line, therefore converting the MPC law into a continuous and **Piecewise-Affine Function** of the parameter vector (Bemporad et al. 2002b).

Agenda

- Explicit & Implicit definition
- Implicit MPC vs Explicit MPC
- Explicit MPC
  - Advantages
  - Multi-parametric programming
  - **Piecewise Affine functions**
  - Methodology
  - Limitations
  - Applications
- Case Study

Definition 1 *We say a function* $L : \Re^m \rightarrow \Re^n$ *is linear if (1) for any vectors* $\mathbf{x}$ *and* $\mathbf{y}$ *in* $\Re^m$, $L(\mathbf{x}+\mathbf{y}) = L(\mathbf{x}) + L(\mathbf{y})$, *and (2) for any vector* $\mathbf{x}$ *in* $\Re^m$ *and scalar* a, $L(a\mathbf{x}) = aL(\mathbf{x})$.

Definition 4 *We say a function* $A : \Re^m \rightarrow \Re^n$ *is affine if there is a linear function* $L : \Re^m \rightarrow \Re^n$ *and a vector* $\mathbf{b}$ *in* $\Re^n$ *such that* $A(\mathbf{x}) = L(\mathbf{x}) + \mathbf{b}$ *for all* $\mathbf{x}$ *in* $\Re^m$.

$$\min_{u_0,\ldots,u_{N-1}} \sum_{k=0}^{N-1} \{||Qx_k||_2 + ||Ru_k||_2\} + ||Px_N||_2$$
$$x_{k+1} = Ax_k + Bu_k,\ k = 0,\ldots,N-1$$
$$x_0 = x(t)$$
$$H_u u_k \le k_u,\ k = 0,\ldots,N-1$$
$$H_x x_k \le k_x,\ k = 0,\ldots,N-1$$
$$H_N x_N \le k_N$$

PWA model

$$\min_{z} \sum_{k=0}^{N-1} \{||Qx_k||_2 + ||Ru_k||_2\} + ||Px_N||_2$$
$$x_{k+1} = f_{PWA}(x_k, u_k),\ k = 0,\ldots,N-1$$
$$x_0 = x(t)$$
$$H_u u_k \le k_u,\ k = 0,\ldots,N-1$$
$$H_x x_k \le k_x,\ k = 0,\ldots,N-1$$
$$H_N x_N \le k_N$$

Agenda

- Explicit & Implicit definition
- Implicit MPC vs Explicit MPC
- Explicit MPC
    - Advantages
    - Multi-parametric programming
    - Piecewise Affine functions
    - **Methodology**
    - Limitations
    - Applications
- Case Study

MPQP

Algorithm 1

1 Let $X \subseteq \mathbb{R}^n$ be the set of parameters (states);
2 execute **partition**(X);
3 for all regions where $z($
a convex set, compute s
4 **end**.

procedure partition(Y
1.1 **let** $x_0 \in Y$ and ϵ th
1.2 **if** $\epsilon \leq 0$ **then exit**;
1.3 For $x = x_0$, compu
the QP (13);
1.4 Determine the set of active constraints when $z = z_0$, $x = x_0$, and build $\tilde{G}$, $\tilde{W}$, $\tilde{S}$;
1.5 If $r = \operatorname{rank} \tilde{G}$ is less than the number ℓ of rows of $\tilde{G}$, take a subset of r linearly independent rows, and redefine $\tilde{G}$, $\tilde{W}$, $\tilde{S}$ accordingly;
1.6 Determine $\tilde{\lambda}(x)$, $z(x)$ from (17) and (18);
1.7 Characterize the CR from (19) and (20);
1.8 Define and partition the rest of the region as in Theorem 3;
1.9 For each new sub-region R_i, **partition**(R_i);
end procedure.

$$\boxed{Hz + G'\lambda = 0,\ \lambda \in \mathbb{R}^q,}$$

$$\lambda_i(G^i z - W^i - S^i x) = 0,\ i = 1, \ldots, q,$$

$$\lambda \geq 0,$$

$$Gz \leq W + Sx,$$

$$\begin{aligned} &\max_{x,z,\epsilon}\ \epsilon \\ &\text{subj. to}\ \ T^i x + \epsilon \|T^i\| \leq Z^i \\ &\qquad\qquad Gz - Sx \leq W \end{aligned} \tag{14}$$

$$\begin{aligned} \min_{U \triangleq \{u_t, \ldots, u_{t+N_u-1}\}} &\Big\{ J(U, x(t)) = x'_{t+N_y|t} P x_{t+N_y|t} \\ &+ \sum_{k=0}^{N_y-1} \left[x'_{t+k|t} Q x_{t+k|t} + u'_{t+k} R u_{t+k} \right] \\ \text{subj. to}\quad & y_{\min} \leq y_{t+k|t} \leq y_{\max},\ k = 1, \ldots, N_c \\ & u_{\min} \leq u_{t+k} \leq u_{\max},\ k = 0, 1, \ldots, N_c \\ & x_{t|t} = x(t) \\ & x_{t+k+1|t} = A x_{t+k|t} + B u_{t+k},\ k \geq 0 \\ & y_{t+k|t} = C x_{t+k|t},\ k \geq 0 \\ & u_{t+k} = K x_{t+k|t},\ N_u \leq k < N_y \end{aligned}$$

$$z = -H^{-1} G' \lambda$$

$$\begin{aligned} V_z(x) = \min_z \quad & \tfrac{1}{2} z' H z \\ \text{subj. to}\ & Gz \leq W + Sx(t), \end{aligned} \tag{13}$$

MPQP

Algorithm 1

1 Let $X \subseteq \mathbb{R}^n$ be the set of parameters (states);
2 execute **partition**(X);
3 for all regions where $z($
a convex set, compute s
4 **end**.

procedure partition(
1.1 **let** $x_0 \in Y$ and ϵ th
1.2 **if** $\epsilon \leq 0$ **then exit**;
1.3 For $x = x_0$, compu
the QP (13);
1.4 Determine the set of active constraints when $z = z_0$, $x = x_0$, and build $\tilde{G}$, $\tilde{W}$, $\tilde{S}$;
1.5 If $r = \operatorname{rank} \tilde{G}$ is less than the number ℓ of rows of $\tilde{G}$, take a subset of r linearly independent rows, and redefine $\tilde{G}$, $\tilde{W}$, $\tilde{S}$ accordingly;
1.6 Determine $\tilde{\lambda}(x)$, $z(x)$ from (17) and (18);
1.7 Characterize the CR from (19) and (20);
1.8 Define and partition the rest of the region as in Theorem 3;
1.9 For each new sub-region R_i, **partition**(R_i);
end procedure.

$$Hz + G'\lambda = 0,\ \lambda \in \mathbb{R}^q,$$

$$\boxed{\lambda_i(G^i z - W^i - S^i x) = 0,\ i = 1, \ldots, q,}$$

$$\lambda \geq 0,$$

$$Gz \leq W + Sx,$$

For inactive constraints, $\breve{\lambda} = 0$.

$$\tilde{\lambda} = -(\tilde{G}H^{-1}\tilde{G}')^{-1}(\tilde{W} + \tilde{S}x) \qquad (17)$$

$$z = -H^{-1}G'\lambda$$

$$z = H^{-1}\tilde{G}'(\tilde{G}H^{-1}\tilde{G}')^{-1}(\tilde{W} + \tilde{S}x) \qquad (18)$$

MPQP

Algorithm 1

1 Let $X \subseteq \mathbb{R}^n$ be the set of parameters (states);
2 execute **partition**(X);
3 for all regions where $z($
a convex set, compute su
4 **end.**

procedure partition$(Y$
1.1 **let** $x_0 \in Y$ and ϵ th
1.2 **if** $\epsilon \leq 0$ **then exit**;
1.3 For $x = x_0$, compu
the QP (13);
1.4 Determine the set of activ
$x = x_0$, and build $\tilde{G}$, $\tilde{W}$, $\tilde{S}$
1.5 If r = rank $\tilde{G}$ is less than
$\tilde{G}$, take a subset of r linea
redefine $\tilde{G}$, $\tilde{W}$, $\tilde{S}$ accordingly;
1.6 Determine $\tilde{\lambda}(x)$, $z(x)$ from (17) and (18);
1.7 Characterize the CR from (19) and (20);
1.8 Define and partition the rest of the region as in Theorem 3;
1.9 For each new sub-region R_i, **partition**(R_i);
end procedure.

$$Hz + G'\lambda = 0, \;\; \lambda \in \mathbb{R}^q,$$

$$\lambda_i(G^i z - W^i - S^i x) = 0, \;\; i = 1, \ldots, q,$$

$$\boxed{\lambda \geq 0,}$$

$$Gz \leq W + Sx,$$

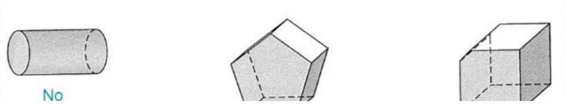
$$V_z(x) = \min_z \;\; \tfrac{1}{2} z' H z$$

$$\text{subj. to } Gz \leq W + Sx(t), \qquad (13)$$

$$z = -H^{-1} G' \lambda$$

$$G H^{-1} \tilde{G}' (\tilde{G} H^{-1} \tilde{G}')^{-1} (\tilde{W} + \tilde{S}x) \leq W + Sx \qquad (19)$$

$$\tilde{\lambda} = -(\tilde{G} H^{-1} \tilde{G}')^{-1} (\tilde{W} + \tilde{S}x) \qquad (17)$$



MP

Algorithm 1

1 Let $X \subseteq \mathbb{R}^n$ be the set of parameters (states);
2 execute **partition**(X);
3 for all regions where $z(x)$ is the same and whose union is a convex set, compute such a union as in [9];
4 **end.**

procedure partition(Y)
1.1 **let** $x_0 \in Y$ and ϵ the solution to the LP (14);
1.2 **if** $\epsilon \leq 0$ **then exit**; (no full dimensional CR is in Y)
1.3 For $x = x_0$, compute the optimal solution (z_0, λ_0) of the QP (13);
1.4 Determine the set of active constraints when $z = z_0$, $x = x_0$, and build $\tilde{G}$, $\tilde{W}$, $\tilde{S}$;
1.5 If $r = \operatorname{rank} \tilde{G}$ is less than the number ℓ of rows of $\tilde{G}$, take a subset of r linearly independent rows, and redefine $\tilde{G}$, $\tilde{W}$, $\tilde{S}$ accordingly;
1.6 Determine $\tilde{\lambda}(x)$, $z(x)$ from (17) and (18);
1.7 Characterize the CR from (19) and (20);
1.8 Define and partition the rest of the region as in Theorem 3;
1.9 For each new sub-region R_i, **partition**(R_i);
end procedure.

Theorem 3
$\{x \in Y : Ax$
Also let

$$R_i = \Big\{ x$$

$$u = \begin{cases}
\begin{bmatrix} -5.9220 & -6.8883 \end{bmatrix} x & \text{if } \begin{bmatrix} -5.9220 & -6.8883 \\ 5.9220 & 6.8883 \\ -1.5379 & 6.8291 \\ 1.5379 & -6.8291 \end{bmatrix} x \leq \begin{bmatrix} 2.0000 \\ 2.0000 \\ 2.0000 \\ 2.0000 \end{bmatrix} \\
 & \text{(Region \#1)} \\
2.0000 & \text{if } \begin{bmatrix} -3.4155 & 4.6452 \\ 0.1044 & 0.1215 \\ 0.1259 & 0.0922 \end{bmatrix} x \leq \begin{bmatrix} 2.6341 \\ -0.0353 \\ -0.0267 \end{bmatrix} \\
 & \text{(Region \#2, \#4)} \\
2.0000 & \text{if } \begin{bmatrix} 0.0679 & -0.0924 \\ 0.1259 & 0.0922 \end{bmatrix} x \leq \begin{bmatrix} -0.0524 \\ -0.0519 \end{bmatrix} \\
 & \text{(Region \#3)} \\
-2.0000 & \text{if } \begin{bmatrix} -0.1259 & -0.0922 \\ -0.0679 & 0.0924 \end{bmatrix} x \leq \begin{bmatrix} -0.0519 \\ -0.0524 \end{bmatrix} \\
 & \text{(Region \#5)} \\
\begin{bmatrix} -6.4159 & -4.6953 \end{bmatrix} x + 0.6423 & \text{if } \begin{bmatrix} -6.4159 & -4.6953 \\ -0.0275 & 0.1220 \\ 6.4159 & 4.6953 \end{bmatrix} x \leq \begin{bmatrix} 1.3577 \\ -0.0357 \\ 2.6423 \end{bmatrix} \\
 & \text{(Region \#6)} \\
-2.0000 & \text{if } \begin{bmatrix} 3.4155 & -4.6452 \\ -0.1044 & -0.1215 \\ -0.1259 & -0.0922 \end{bmatrix} x \leq \begin{bmatrix} 2.6341 \\ -0.0353 \\ -0.0267 \end{bmatrix} \\
 & \text{(Region \#7, \#8)} \\
\begin{bmatrix} -6.4159 & -4.6953 \end{bmatrix} x - 0.6423 & \text{if } \begin{bmatrix} 6.4159 & 4.6953 \\ 0.0275 & -0.1220 \\ -6.4159 & -4.6953 \end{bmatrix} x \leq \begin{bmatrix} 1.3577 \\ -0.0357 \\ 2.6423 \end{bmatrix} \\
 & \text{(Region \#9)}
\end{cases}$$

Agenda

- Explicit & Implicit definition
- Implicit MPC vs Explicit MPC
- Explicit MPC
  - Advantages
  - Multi-parametric programming
  - Piecewise Affine functions
  - Methodology
  - **Limitations**
  - Applications
- Case Study

Limitations

- Though Explicit MPC is faster and uses off-line approach it is capable of handling smaller systems (generally with variables <6).
- Requires large storage units.
- Sequential and less dynamic.

Agenda

- Explicit & Implicit definition
- Implicit MPC vs Explicit MPC
- Explicit MPC
  - Advantages
  - Multi-parametric programming
  - Piecewise Affine functions
  - Methodology
  - Limitations
  - **Applications**
- Case Study

Applications

- In literature, it can be seen that Explicit MPC techniques have addressed several industrial problems.
- Explicit MPCs are reported to be most suitable for fast-sampling problems (in the order of 1-50 ms) and relatively small size (1-2 manipulated inputs, 5-10 parameters) such as **automotive domain and electrical power converters**.

Agenda

- Explicit & Implicit definition
- Implicit MPC vs Explicit MPC
- Explicit MPC
  - Advantages
  - Multi-parametric programming (Linear and Quadratic)
  - Piecewise Affine functions
  - Methodology (MPLP and MPQP
  - Limitations
  - Applications
- **Case Study**

Conclusion

- Explicit MPC is a powerful tool to convert an MPC design into an equivalent offline control law that has simpler implementations.
- Implementation of the explicit MPC for solving the QP problem depends on available CPU time, data memory, and program memory and other practical considerations.
- Explicit MPC approach remains convenient for relatively small and offline problems (such as one or two command inputs, short control and constraint horizons, up to ten states).
- For larger problems, and/or problems that are linear time varying, online QP solution methods tailored to conventional Implicit MPC may be preferable.

References

1) Model Predictive Control nased on Linear Programming –The Explicit Solution, 2001
2) Online optimization via offline parametric optimization tools, 2000
3) The Explicit Linear Quadratic Regulator for Constrained Systems, 2001
4) Explicit Model Predictive Control, 2014
5) Explicit MPC: Basics, Fast implementations, advantages and limitations
6) https://www.mathworks.com/help/mpc/controller-design.html
7) Borrelli, F., Bemporad, A., Fodor, M., Hrovat, D.: A hybrid approach to traction control. In: Di Benedetto, M.D., Sangiovanni-Vincentelli, A.L. (eds.) HSCC 2001. LNCS, vol. 2034, pp. 162–174. Springer, Heidelberg (2001)

Simulation Results