Nonlinear MPC via Novel Multiparametric Programming Techniques

Diogo Narciso
d.narciso@imperial.ac.uk

Stratos Pistikopoulos
e.pistikopoulos@imperial.ac.uk
Centre for Process Systems Engineering (CPSE), Imperial College London

Outline

- Framework and objectives
- Nonlinear Multiparametric discrete-time MPC
  - Formulation & overall strategy
  - Off-line strategy
  - Parametric Algorithm
  - On-line strategy
  - Research and Development Achievements
- Concluding remarks and future work
Framework and objectives

- High fidelity model
  - Model Reduction
    - Reduced model
      - Multi-parametric programming
        - MPC
          - Multi-parametric controller

Solution Validation

Framework and objectives

- Model Reduction
  - Reduces model complexity/size

- Multiparametric programming
  - Reduces online control effort

Reduce computational effort or simulation/optimisation time
Nonlinear Multiparametric discrete-time MPC

- Formulation & overall strategy
- Off-line strategy
- Parametric Algorithm
- On-line strategy

Formulation & overall strategy

**Traditional MPC:**

\[
\begin{align*}
\text{min} & \quad \text{Objective Function } (U, x(t), z(t)) \\
\text{s.t.} & \quad \text{Dynamic Model } (U, x(t), z(t)) \\
& \quad \text{Process Constraints } (U, x(t), z(t)) \\
& \quad \text{Initial Conditions}
\end{align*}
\]

- Requires specification of initial conditions
- Optimal inputs calculated for the vector of initial conditions

**Multiparametric MPC:**

\[
\begin{align*}
\text{min} & \quad \text{Objective Function } (U, \theta) \\
\text{s.t.} & \quad \text{Dynamic Model } (U, \theta) \\
& \quad \text{Process Constraints } (U, \theta) \\
& \quad \text{Parameter Space Definition}
\end{align*}
\]

- States/disturbances recast as parameters \( \theta \)
- The space of expected parameters defined as an independent space
- Problem is solved for all combinations of parameters (semi-infinite problem)
Formulation & overall strategy

- MPC Formulation
- Multiparametric MPC Formulation

Recast:
- User defines parameters
- User defines parameter space

Off-line strategy

- MPC Formulation
- Multiparametric MPC Formulation

Advantages:
- Speed;
- Accuracy;

States (Parameters)

Critical Regions

Optimal inputs

Optimal inputs

U(\theta) = \omega_i + W_i \theta

CR: \phi_i \theta \leq \phi_i

U(\theta) = \omega_i + W_i \theta

CR: \phi_i \theta \leq \phi_i

Off-line strategy

- MPC Formulation
- Multiparametric MPC Formulation

Process (Parameters)

Multiparametric MPC Solution
Off-line strategy

1) Select parameters
   1) States
   2) Load disturbances

2) Define parameter space
   1) e.g. Reactor temperature between 25 and 75°C

3) Discretise time domain
   1) Need to express state constraints with finite number of equations

4) Discretise controls and load disturbances
   1) Control vector parameterization

\[
\min_{U} \text{Objective Function (} U, x(t), z(t) \text{)}
\]
\[s.t. \text{ Dynamic Model (} U, x(t), z(t) \text{)}\]
\[\text{Process Constraints (} U, x(t), z(t) \text{)}\]
\[\text{Initial Conditions}\]

\[
\min_{U} \text{Objective Function (} U, \theta \text{)}
\]
\[s.t. \text{ Dynamic Model (} U, \theta \text{)}\]
\[\text{Process Constraints (} U, \theta \text{)}\]

Parameter Space Definition
Off-line strategy

5) Assess constraints
   1) Formulate candidate point constraints
   2) Solve relaxed optimisation problem
   3) Add new constraints to mp-NLP problem

\[
\begin{align*}
\min_{U, \theta} & \quad \text{Objective Function} (U, \theta) \\
\text{s.t.} & \quad \text{Dynamic Model} (U, \theta) \\
& \quad \text{Process Constraints} (U, \theta) \\
& \quad \text{Parameter Space Definition}
\end{align*}
\]

- Vector of inputs (finite)
- Inputs parameterized
- States discretised in time
- Vector of constraints (finite)
Parametric Algorithm

- Driven by accurate active-set characterization of parameter space
- Vertex search method
- Valid for convex formulations
Parametric Algorithm

- Components of the new algorithm
  - Search of vertices
  - Construction of critical regions
  - Approximation of optimal solutions

Problem formulation:

\[
\begin{align*}
\min_{x_1, x_2} & \quad x_1^3 + 2x_1^2 - 5x_1 + 6x_2^2 - 3x_2 - 6 \\
\text{s.t.} & \quad 2x_1 + x_2 \leq 2.5 + \theta_1 \\
& \quad 0.5x_1 + x_2 \leq 1.5 + \theta_2 \\
& \quad x_1 \geq 0 \\
& \quad x_2 \geq 0 \\
& \quad 0 \leq \theta_1 \leq 1 \\
& \quad 0 \leq \theta_2 \leq 1
\end{align*}
\]
Parametric Algorithm

**Stage 1: Vertex search**

“A point is called a vertex if a total number of boundaries and/or parametric walls equal to the size of the space of parameters is active in the context of a specific critical region.”
Parametric Algorithm

Stage 1: Vertex search

- The vertices are just points of the parameter space!
- How to use this information to approximate the critical regions?
Parametric Algorithm

Stage 2: Boundary construction

- We assess all the active sets in the vicinity of each of the found vertices

<table>
<thead>
<tr>
<th>Vertex</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active</td>
<td>1</td>
<td>∅</td>
<td>1</td>
<td>∅</td>
<td>2</td>
<td>1&amp;2</td>
<td>1</td>
<td>∅</td>
<td>1&amp;2</td>
</tr>
<tr>
<td>Constraint</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1&amp;2</td>
<td>1&amp;2</td>
<td>1&amp;2</td>
<td>1&amp;2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- This information enables one to derive a list of all existing boundaries between active sets, based on the solution tree:

```
∅
/
/  \
1-2
/
/  \
1&2
```

Parameter space ($\Theta$):

Found Vertices

```
1 2 3 4 5 6 7 8 9
CR₂ CR₁ CR₄ CR₃
θ₁ θ₂
```

Parametric Algorithm

Stage 2: Boundary construction

- We take the vertices found for each boundary and create linear approximations, based on the defined solution tree
- The accuracy is assessed
Parametric Algorithm

Stage 3: Parameterization of optimal solutions

- Convex hulls
- Linear Interpolation
- Assess the error
- Create partitions

**Parametric Algorithm**

**Final Solution:** Map of critical regions:
On-line strategy

MPC Formulation

Multiparametric MPC Formulation

Process (Parameters) ↔ Optimal inputs ↔ Multiparametric MPC Solution

States

Measure

Space of states

$x_1$

$x_2$

$U(\theta)_i = \omega_i + W_i \theta$

$CR_i: \Phi_i \theta \leq \phi_i$

Optimal inputs

Multiparametric MPC Solution
On-line strategy

- Performs only linear calculations
- Consists of an input-output methodology based on the multiparametric solution

\[
CR_i: \Phi_i \theta \leq \phi_i
\]

\[
U(\theta)_i = \omega_i + W_i \theta
\]

Research and Development Achievements

- Theoretical framework completed
  - On-line strategy
  - Parametric Algorithm
  - Off-line strategy
- Software has been developed (C++/gPROMS)
  - Parametric Solver (Implementation of Algorithm)
  - Off-line strategy
Concluding remarks

- A novel framework for the use of Multiparametric MPC has been developed;
- Theoretical and practical developments have been made for each of its components;
- Novel developments on the Parametric Solver will enable to solve more complex problems;
- Small chemical engineering problem solved;
- Further testing will enable to improve the methodology.

Example

- $F^1$ – control stream
- $F^2$ – load disturbance
- $F^3$ – outlet stream (gravity)
- Problem defined for a control horizon of 3 time units
Example

Problem formulation:

\[
\begin{align*}
\min_{F_0^1,F_0^2,F_0^3} \quad & P(h_3 - h^{op})^2 + \sum_{i=0}^{2} Q(h_i - h^{op})^2 + R(u_{i+1} - u_i)^2 \\
\text{s.t.} \quad & h_{i+1} = h_i + D_i \Delta t, \quad l \geq 0 \\
& D_i = \frac{F_i^1 + F_i^2 - C_D \sqrt{h_i}}{A}, \quad l \geq 0 \\
& h_0 = h_0, \quad u_0 = u_0; \quad \begin{pmatrix} F_2^1 & F_2^2 & F_2^3 \end{pmatrix}^T = (\theta_1 \theta_2 \theta_3)^T; \\
& h_{\min} \leq h_r \leq h_{\max}, \quad r \geq 1; \\
& \begin{pmatrix} 0 & 0 & 0 \end{pmatrix}^T \leq \begin{pmatrix} F_1^1 & F_1^2 & F_1^3 \end{pmatrix}^T \leq \begin{pmatrix} 1.5 & 1.5 & 1.5 \end{pmatrix}^T \\
& \begin{pmatrix} 0 & 0 & 0 \end{pmatrix}^T \leq (\theta_1 \theta_2 \theta_3)^T \leq \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}^T
\end{align*}
\]

Example of optimal solution

\[ (\theta_1 \theta_2 \theta_3)^T = (1 1 1)^T \]

Final Solution:

Parametric solutions for Critical Region 1:

\[
\begin{align*}
F_1^1 &= 1.029 - 0.284\theta_1 - 0.278\theta_2 - 0.010\theta_3 \\
F_2^1 &= 1.385 - 0.473\theta_1 - 0.466\theta_2 - 0.019\theta_3 \\
F_3^1 &= 1.461 - 0.513\theta_1 - 0.509\theta_2 - 0.025\theta_3
\end{align*}
\]
Reduction of the control/optimisation model

- Given a linear state space, we seek to decrease the number of states.

Balanced Truncation - fundamentals

- From a dynamics point of view, we want to neglect the states which are harder to reach and harder to observe.
- Using a balanced realization, to reduce a system from order $n$ to $n-p$, we neglect the last $p$ states.
Reduction of the control/optimisation model

- Order of reduction, $p$, is defined (size = $n$)
  - $x_1$ – first $n-p$ components of $x$
  - $x_2$ – last $p$ components of $x$
- Relevant partitions of the states vector and matrices are made

$$
\min J(U, x(t)) = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_t^T \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_{t+N_{yt}} + \sum_{k=0}^{N_y-1} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_{t+k+N_{yt}}^T \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_{t+k+N_{yt}} + u_{{t+k}}^T R u_{{t+k}}
$$

$s.t.$
$$y_{min} \leq y_{t+k|t} \leq y_{max}, \ k = 1, ..., N_c$$
$$u_{min} \leq u_{t+k} \leq u_{max}, \ k = 0, 1, ..., N_c$$

\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_{t+k|t} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_{t+k|t} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u_{t+k}, \ k \geq 0$$

$$y_{t+k|t} = \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_{t+k|t}, \ k \geq 0$$

$$u_{t+k} = \begin{bmatrix} K_1 \\ K_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_{t+k|t}, \ N_u \leq k \leq N_y$$

Combined Balanced Truncation/ mp control – framework

- Iterate with order reduction
- Objective: find the minimum order reduction for which feasibility and optimality are guaranteed

[Diagram showing the decision process for full size mp-MPC versus reduced mp-MPC, with decision points for feasibility, infeasibility, dynamic performance, and meeting the need, leading to increase order reduction, resize output, resize input, or reduce order reduction.]
Illustrative example

- Full size closed loop response is close to open loop response
- Reasonable performance for small order reductions
- Appropriate control design is key!

Optimal Model Reduction order: from 30 to 20 states