Globally Optimal Nonlinear Model Predictive Control

C. E. Long, P. K. Polisetty, and E. P. Gatzke
Department of Chemical Engineering, University of South Carolina

Presentation Outline

• Background and Motivation
• Nonlinear Model Predictive Control
  ➢ Formulation
  ➢ Deterministic Method for Solving Nonconvex Problem
• Case Study
  ➢ Isothermal CSTR with Van de Vusse kinetics
• Summary
Model Predictive Control (MPC)  
(Garcia, Prett, and Morari, 1989)

- Model Predictive Control is an advanced control algorithm that handles:
  - Multivariable interacting systems
  - Soft constraints on both inputs and outputs
  - Measured disturbances
  - Process time delays and difficult dynamics
  - Known future setpoint changes (reference transitions)
- Limitations
  - Requires an explicit model of the process
  - Need to solve a constrained optimization problem online

2. Solves the problem online and implements the optimal control move.
3. Waits for new data at next time step.
Linear vs. Nonlinear Formulations

**Linear Formulations:**
- Mismatch between linear model and nonlinear process can lead to poor closed-loop performance and/or instability.
- Linear model (linear constraints) -> convex problem (only one minimum)
- Efficient solvers (local) exist to solve the convex problem

**Nonlinear Formulations:**
- Nonlinear model handles process nonlinearities.
- Nonlinear model (nonlinear constraints) -> nonconvex problem (multiple minima)
- Local solvers left susceptible to choosing suboptimal minima.
- Need to guarantee global optimality in nonconvex problems.
Nonlinear Model Predictive Control (NMPC) Formulation

Objective Function:
\[
\Phi = \sum_{i=1}^{p} \Gamma_e e(i) + \sum_{i=1}^{m} \Gamma_u \Delta u(i)
\]

Constrained Optimization Problem:
\[
\min_{\{u(i)\}, \ u(m)} \Phi
\]

subject to:
\[
x(k+1) = f(x(k), u(k))
\]
\[
y(k) = g(x(k), u(k))
\]
\[
|r(i) - y(i)| \leq e(i) \quad \forall i = 1 \ldots p
\]
\[
|u(i-2) - u(i-1)| \leq \Delta u(i-1) \quad \forall i = 1 \ldots m
\]
\[
d(i) = y_m(0) - y_p(0)
\]
\[
u^L \leq u \leq u^U
\]
Online vs. Offline Methods for MPC

**Online**
- Formulate the appropriate optimization problem based on current data at each time step.
- Solve the problem online.
  - computationally demanding
  - real-time constraints
- Possibility of unnecessarily solving the same problem over and over again
- Solution is optimal and pertains exactly to the process’ current state.

**Offline**
- Partition solution space into characteristic regions based on a set of parameters (states, inputs, etc…)
- Solve problems from each region offline
  - high dimensionality issues
  - Can you foresee all scenarios?
- At each time step, identify appropriate region online based on current data and “look up” the solution.
  - low online computational demand
- Implement solution from the region.
  - Suboptimal?
Deterministic Method for Solving the Nonconvex NLP

\[
\begin{align*}
\min_{x} & \quad C^T x \\
\text{s.t.} \quad & \quad A x \leq b \\
& \quad f(x) = 0 \\
& \quad x^L \leq x \leq x^U
\end{align*}
\]

Reformulation

Create Convex Relaxations

\[
\begin{align*}
\min_{w,x} & \quad C^T x \\
\text{s.t.} \quad & \quad A x \leq b \\
& \quad A_2 \left( \frac{w}{x} \right) = 0 \\
& \quad A_3 \left( \frac{w}{x} \right) \leq b_3 \\
& \quad x^L \leq x \leq x^U \\
& \quad w^L \leq w \leq w^U
\end{align*}
\]

Linearization of Convex Functions

Branch and Bound Tree

Solve LP

Fathoming

Branch the Space

Consider Each Partition as a New Problem

\[
\begin{align*}
\bar{g}(w,x,x^L,x^U,w^L,w^U) \leq w \leq \bar{g}(w,x,x^L,x^U,w^L,w^U)
\end{align*}
\]
Benchmark Control Problem

Consider the isothermal operation of a SISO two state CSTR exhibiting Van de Vusse kinetics:

\[ \begin{align*}
A \xrightarrow{k_1} B \xrightarrow{k_2} C \\
2A \xrightarrow{k_3} D
\end{align*} \]

Where the corresponding reaction rates are:

\[ \begin{align*}
\dot{r}_A &= -k_1 C_A - k_3 C_A^2 \\
\dot{r}_B &= k_1 C_A - k_2 C_B
\end{align*} \]

The system can be described by:

\[ \begin{align*}
\frac{dC_A}{d} &= \left(\frac{F}{V}\right)(C_{A,0} - C_A) - k_1 C_A - k_3 C_A^2 \\
\frac{dC_B}{d} &= k_1 C_A - k_2 C_B - \left(\frac{F}{V}\right)C_B
\end{align*} \]

where:

\[ \begin{align*}
C_A &= \text{Conc. of Species A} \\
C_B &= \text{Conc. of Species B} \\
k_1 &= \text{Reaction Rate Constants} \\
F &= \text{Feed Flow Rate} \\
V &= \text{Reactor Volume (constant)} \\
C_{AO} &= \text{Conc. of A in the Feed} \\
\end{align*} \]
Closed-loop Performance Test

Both setpoint tracking and disturbance rejection are tested through a series of setpoint transitions and disturbance loads.

Assume the process is initially being operating where:

- Feed Flow Rate/Reactor Volume (F/V) = 181 mol/liter-hr
- Conc. of B (C_B) = 1.1 mol/liter
- Conc. of A in the Feed (C_AO) = 10 mol/liter

\[ y_{sp} = 1.1 \]

\[ y_{sp} = 1.0 \]

\[ C_{AO} = 10 \]

\[ C_{AO} = 9 \]

\[ C_{AO} = 7 \]

\[ y_{sp} = 0.8 \]

\[ t=0 \quad t=0.1 \quad t=0.2 \quad t=0.3 \quad t=0.4 \quad t=0.5 \quad t=0.6 \quad \text{hrs} \]
Closed-loop Results
(single degree of freedom)

Objective Function:
\[ \Phi = \Gamma_y e(p) + \sum_{i=1}^{m} \Gamma_u \Delta u(i) \]

Tunings:
- \( m = 1 \)
- \( \Gamma_y = 100 \)
- \( p = 30 \)
- \( \Gamma_u = 0 \)

Steady State Loci for the Reactor Operation at Different Feed Concentrations \( (C_{AO}) \) Exhibiting the Presence of an Input Multiplicity
Sample Optimization Problem

Assume the process is operating at:

\[ u = 181 \text{ mol/liter-hr} \]
\[ y = 1.1 \text{ mol/liter} \]
\[ C_{AO} = 10 \text{ mol/liter} \]

Consider the Objective Function as

\[ \Phi = \Gamma_y e(p) + \sum_{i=1}^{m} \Gamma_u \Delta u(i) \]

(terminal weight)

Let:

\[ m = 1 \]
\[ \Gamma_y = 100 \]
\[ p = 30 \]
\[ \Gamma_u = 0 \]

Assume bounds on the input.

\[ 0 \leq u \leq 200 \]

Sample objective function for problem having the setpoint moved from the initial condition of 1.1 mol/liter to 1 mol/liter.
Real-Time Considerations

- Sampling Time = 7.2 seconds
- Must terminate solves to meet real-time constraints.
- Global solution might have been found, not guaranteed.

Sample objective function for a problem that takes longer than 7.2 seconds to solve globally.
Closed-loop Results
(multiple degrees of freedom)

Objective Function:
\[ \Phi = \sum_{i=1}^{P} \Gamma_y e(i) + \sum_{i=1}^{m} \Gamma_u \Delta u(i) \]

Tunings:
- \( m = 2 \)
- \( \Gamma_y = 100 \)
- \( p = 30 \)
- \( \Gamma_u = 0.005 \)

Sample objective function at the time of the first setpoint change for the NMPC with two degrees of freedom (m=2)
Use of a Terminal Weight
(multiple degrees of freedom)

Objective Function:

\[ \Phi = \Gamma_y e(p) + \sum_{i=1}^{m} \Gamma_u \Delta u(i) \]

Tunings: \( m = 2 \) \( \Gamma_y = 100 \)
\( p = 30 \) \( \Gamma_u = 0.005 \)

As in the single degree of freedom (m=1) case, the modified objective function (using the terminal weight) allows for the controller to better track the setpoint.
Incorporating Hard Constraints

Objective Function: \( \Phi = \Gamma_y e(p) + \sum_{i=1}^{m} \Gamma_u \Delta u(i) \)

Tunings: \( m = 2 \quad \Gamma_y = 100 \quad e(p) = 0 \)

\( p = 30 \quad \Gamma_u = 0.005 \)

Possibility for Infeasible Problems
(constraint relaxation may be necessary!)

Problem feasibility when a hard constraint is imposed on \( e(p) \). A value of 0 indicates feasible problems, while a value of 1 shows infeasible problems in which a hard constraint relaxation is necessary.
Traditional Objective Function with Hard Constraint

Objective Function:
\[ \Phi = \sum_{i=1}^{p} \Gamma_y e(i) + \sum_{i=1}^{m} \Gamma_u \Delta u(i) \]

Tunings: 
- \( m = 2 \) 
- \( \Gamma_y = 100 \) 
- \( p = 30 \) 
- \( \Gamma_u = 0.005 \) 
- \( e(p) = 0 \)

Improvement over using the traditional objective function without the hard constraint.
Noise Run

Objective Function
\[ \Phi = \sum_{i=1}^{n-1} \Gamma_y e(i) + \Gamma_p e(p) + \sum_{i=1}^{m} \Gamma_u \Delta u(i) \]

- \( e(p) \) weighted more heavily than other errors.

Tunings
- \( m = 2 \)
- \( \Gamma_y = 100 \)
- \( p = 30 \)
- \( \Gamma_u = 0.005 \)
- \( \Gamma_t = 10,000 \)

Noise
- white measurement noise with a standard deviation of 3%
Summary

- A NMPC algorithm using a deterministic global optimization search method is proposed.

- The deterministic approach guarantees global optimums to the nonconvex NLPs associated with the NMPC formulation.

- The algorithm eliminates poor performance in the CSTR example resulting from suboptimal input trajectories provided by local solution techniques.

- The proper objective function and controller tunings must be utilized to achieve the desired closed-loop results.

- Considerations must be made for cases where the desired solution cannot be obtained sufficiently fast for real-time use.

Acknowledgements:
ACS PRF #38539-G9
NSF Early Career Development #CTS-0238663