A Multi-objective Model Predictive Control for Temperature Control in Extrusion Processes

Jingyi Lu¹, Ridong Zhang², Ke Yao¹ and Furong Gao¹,²,*

Abstract—In this paper, we consider the temperature control problem of an extrusion process with both heaters and coolers. For the purpose of energy saving and avoiding frequent switch between the heaters and coolers, the coolers are only used to guarantee the barrel temperature below a given safety bound. When this safety constraint is satisfied, the heaters take actions for reference tracking. This scheme is formulated as a multi-objective optimization problem in the framework of model predictive control. Different objectives have different priority. The safety constraint is of the highest priority, and formulated as a constraint in the optimization. Minimization of the inputs corresponding to the cooler is of the second priority, and incorporated into the objective function with heavy penalty weight. Minimization of the tracking error is of the lowest priority. Thus, this term is also incorporated into the objective function, but with light penalty weight. In this way, energy consumption can be reduced and frequent switch can be avoided. Moreover, a polytopic invariant set is developed to guarantee recursive feasibility of the proposed MPC. Simulations are also conducted to show the effectiveness of the proposed method.

I. INTRODUCTION

Extrusion [1], [2] is an important method in polymer processing industry. It is widely applied to produce polymer products. In this process, raw polymers are fed into the extruder through a hopper, and then heated and become molten in the barrel. Rotation of the screw forces the molten materials into a die which forms the materials into the desired shapes. During the heating process, uniform barrel temperature can help to avoid viscosity fluctuations and thermal degradation, which may significantly impact product quality [3], [4], [5]. Therefore, a good temperature control is necessary for the polymer extrusion process.

An extruder is often equipped with both electrical heaters and coolers. Heaters are arranged along the barrels and used to warm up the raw polymers and maintain the temperature at a given setpoint. In addition to the heat generated by the heaters, a large amount of heat is generated by the shearing due to the screw rotation. Screw rotation can produce sufficient, sometimes even excessive heat for the process. For this reason, cooling rather than heating is necessary. However, this dual input arrangement may cause two issues: (1) frequent switch between the two; (2) simultaneous activation of heating and cooling, leading to extra energy cost. In order to save energy and avoid frequent switch, the coolers, including cooling water and cooling fans, are often considered to be auxiliary for temperature control in an extrusion process. They are used to ensure the barrel temperature below a safety bound when excessive heat is generated by the shearing. That is to say, the coolers keep to be idle for most of the time and only turned on when the temperature is about to exceed the safety bound.

In the framework of model predictive control (MPC [6]), the above mentioned design requirements can be formulated as a multi-objective optimization problem. In order of importance, the objectives include: (1) guaranteeing the temperature below the safety bound; (2) minimizing the use of the cooler; (3) minimizing the tracking error. A straightforward method to solve this problem is to assign heavier penalty weight on the objectives with lower priority. This is a commonly adopted method in multi-objective MPC [7], [8]. However, this method is heuristic and the control performance is greatly affected by the penalty weight. Instead of assigning different penalty weight, in [9], a prioritized multi-objective MPC was solved by a lexicographic method, which separates the original problem into a sequence of single-objective optimization. However, the computation becomes intense. In [10], a lasso model predictive control was proposed. The authors provided a systematic way on selecting the penalty weight. The method can guarantee that when the state stays inside a given set, the inputs with lower priority are kept to be zero. However, such a method cannot guarantee that the state always stays inside the given set. Therefore, the method is not suitable for our problem.

In this paper, specific to the temperature control problem in an extrusion process, we propose a multi-objective model predictive control. Firstly, we incorporate the safety bound into the constraint. This endows the safety constraint the highest priority. Then, both of the tracking errors and the inputs corresponding to the coolers are minimized in the objective function. We assign a significantly high penalty weight on the inputs corresponding to the coolers. In this way, the coolers will only take actions when shutting down the heaters cannot make the temperature below the given safety bound. In this control scheme, satisfaction of the design requirements depends on the feasibility of the optimization problem. Therefore, we further design a polytopic invariant set to guarantee the recursive feasibility of the optimization problem.

The remaining of the paper is arranged as follows: Section 2 gives the formulation of the problem; Section 3 provides details of the controller design; Section 4 shows simulation results of the proposed method on an extrusion process with four temperature zones; Finally, Section 5 draws the...
II. PROBLEM FORMULATION

An extrusion process with both coolers and heaters is modeled in the state space as
\[ x(t + 1) = Ax(t) + B_1 u_1(t) + B_2 u_2(t) + d(t). \]  

(1)

Assume the barrel can be divided into \( n \) zones. Then, the vectors \( x, u_1, u_2, d \in \mathbb{R}^n \). Each coordinate of \( x, u_1, u_2 \) denotes the barrel temperature, opening of the heater and opening of the cooler in each zone. \( d \) denotes the temperature variation caused by the shear stress. The actual value of \( d(t) \) is unknown, but it is known that it is bounded in a polytopic set as
\[ d(t) \in \Phi, \quad \forall t > 0. \]

In addition, constraints are posed on the states and inputs as
\[ x(t) \in X, \quad u_1(t) \in U_1, \quad u_2(t) \in U_2 \]
for all \( t \). The set \( X, U_1 \) and \( U_2 \) are all polytopic sets. For simplicity, the model in (1) can be written in a more compact manner as
\[ x(t + 1) = Ax(t) + Bu(t) + d(t), \]
with
\[ B = \begin{bmatrix} B_1 & B_2 \end{bmatrix}, \quad u(t) = \begin{bmatrix} u_1^T(t) & u_2^T(t) \end{bmatrix}^T. \]

Denote the given tracking reference as \( x_r \) with \( x_r \in \mathbb{R}^n \), and the allowed safety bound of the state \( x \) as \( x_u \). The primary objective of the controller design is to guarantee
\[ x \in X_u = \{ x : x \leq x_u \}. \]

Here the inequalities are defined in coordinate wise. When this objective is satisfied, it is desired that \( x \) should track the given reference \( x_r \) by only tuning the heaters. Based on this scheme, a multi-objective model predictive control is proposed in the following section.

III. CONTROLLER DESIGN

A. Prediction model

In order to develop an offset free MPC [11], the prediction model is derived based on (2) by taking difference of the states between two consecutive time instances as
\[ \Delta x_p(t + 1) = A \Delta x(t) + B \Delta u_p. \]  

(3)

Here \( \Delta \) is a time-wise backward difference operator defined as
\[ \Delta f(t) = f(t) - f(t - 1) \]
for any given vector \( f \), \( x_p \) denotes the predicted states and \( u_p \) denotes the predicted control inputs. The equation in (3) enables us to further derive that
\[ x_p(t + 1) = \Delta x_p(t + 1) + x(t). \]

Then, a multi-step prediction model with prediction horizon taken as \( p_n \) can be derived by
\[ \begin{bmatrix} \Delta x_p(t + i + 1) \\
 x_p(t + i + 1) \end{bmatrix} = \begin{bmatrix} A & 0 \\
 A & I \end{bmatrix} \begin{bmatrix} \Delta x_p(t + i) \\
 x_p(t + i) \end{bmatrix} + \begin{bmatrix} B \\
 B \end{bmatrix} \Delta u_p(t + i) \]
with \( i = 0, 1, \ldots, p_n \) and
\[ \Delta x_p(t) = x(t) - x(t - 1) \quad x_p(t) = x(t). \]

If we denote
\[ \bar{A} = \begin{bmatrix} A & 0 \end{bmatrix}, \quad \bar{B} = \begin{bmatrix} B \end{bmatrix}, \quad \bar{C} = \begin{bmatrix} 0 & I \end{bmatrix}, \]
and
\[ x_0(t) = \begin{bmatrix} x^T(t) - x^T(t - 1) & x^T(t) \end{bmatrix}, \]
\[ X_p(t + 1) = \begin{bmatrix} x_p(t + 1)^T & x_p(t + 2)^T & \cdots & x_p(t + p_n)^T \end{bmatrix}^T, \]
\[ \Delta U_p(t) = \begin{bmatrix} \Delta u_p(t)^T & \Delta u_p(t + 1)^T & \cdots & \Delta u_p(t + p_n - 1)^T \end{bmatrix}^T, \]
it can be derived that
\[ X_p(t + 1) = \bar{A} x_0(t) + \bar{B} \Delta U_p(t) \]
and
\[ U_p(t) = I_{a} \Delta U_p(t) + \begin{bmatrix} u^T(t) & 0^T & \cdots & 0^T \end{bmatrix}. \]  

(5)

Here
\[ \bar{A} = \begin{bmatrix} C\bar{A} & C\bar{A}^2 & \cdots & C\bar{A}^{p_n} \\
 \vdots & \ddots & \vdots & \ddots \\
 C\bar{A}^{p_n} & \cdots & C\bar{A}^{p_{n-1}} & C\bar{A}B & CB \\
 \end{bmatrix}, \quad \bar{B} = \begin{bmatrix} CB & \cdots & CB \\
 C\bar{A}B & \cdots & CB \\
 \vdots & \ddots & \vdots & \vdots \end{bmatrix} \]
and
\[ I_{a} = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\
 -1 & 1 & 0 & \cdots & 0 \\
 0 & -1 & 1 & \cdots & 0 \\
 \vdots & \vdots & \vdots & \ddots & \vdots \end{bmatrix} \in \mathbb{R}^{p_n \times p_n}. \]

B. Controller design

Define a reference vector at \( t + 1 \) as
\[ x_r(t + 1) = \begin{bmatrix} x_r(t + 1)^T & \cdots & x_r(t + p_n)^T \end{bmatrix}. \]

A multi-objective MPC can be designed based on the prediction model in (4) by a convex optimization as
Problem 1:

\[
\min_{\mathbf{U}_p(t)} J(t)
\]
\[
s.t. (4), (5)
\]
\[
x_p(t+i) \in \mathcal{X}_u \\
u_1(t+i-1) \in \mathcal{U}_1 \\
u_2(t+i-1) \in \mathcal{U}_2 \\
x_p(t+i) \in \mathcal{X}, i = 1, 2, \ldots, p
\]

The objective function is defined as

\[
J(t) = [\mathbf{X}_r(t+1) - \mathbf{X}_p(t+1)]^T Q [\mathbf{X}_r(t+1) - \mathbf{X}_p(t+1)] + \Delta \mathbf{U}_p(t)^T R \Delta \mathbf{U}_p(t) + \| S \mathbf{U}_p(t) \|_1. 
\]

Here
\[
Q \in \mathbb{R}^{np_n \times np_n}, R \in \mathbb{R}^{2np_n \times 2np_n}, Q \text{ and } R \text{ are positive definite matrices.}
\]
In general, \(Q\) is taken to be an identity matrix \(I\). The matrix \(R\) is used to regularize the input term. Thus, it is set to be \(R = \epsilon I\) with \(\epsilon\) a small positive scalar. The last term in \(J(t)\) is used to assign different priority on the heaters and coolers. Here the matrix \(S\) is defined as

\[
S = \begin{bmatrix}
\Lambda & 0 & 0 & \cdots & 0 \\
0 & \Lambda & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & \cdots & 0 \\
0 & 0 & \cdots & 0 & \Lambda
\end{bmatrix} \in \mathbb{R}^{2np_n \times 2np_n},
\]

with

\[
\Lambda = \begin{bmatrix}
0 & 0 & \cdots \\
0 & 0 & 0 \\
0 & \cdots & \gamma \mathbf{1}^T \times n
\end{bmatrix}.
\]

Here \(\gamma \gg 1\). In this way, heavy penalty is put on the predicted input term corresponding to the coolers.

Due to this heavy penalty weight, the coolers will be kept turned off if the safety constraint can be satisfied by only adjusting the heater. In this way, we assign a higher priority on the safety constraint and the heaters. The coolers have the lowest priority for the sake of energy saving. In addition, since the safety bound is usually a bit larger than the tracking reference, the gap between the safety bound and the reference can be considered as a deadzone. This deadzone can help to reduce switch between the two inputs.

Compared with the method in [10], the proposed method do not need to carefully select the weight matrix \(S\). Here we only need to take a \(S\) with large enough \(\gamma\). Moreover, the proposed method can guarantee the satisfaction the safety constraint if the optimization in Problem 1 is feasible. Thus, the critical issue is how to ensure the consistent feasibility of the optimization.

C. Recursive feasibility

Here we use recursive feasibility to guarantee consistent feasibility. In order to guarantee recursive feasibility of the optimization in Problem 1, a polytopic invariant set \(\Omega\) on the state \(x\) is designed. The set \(\Omega\) should satisfy that

\[
\text{if } x \in \Omega, \text{ then } x \in \mathcal{X}_u \text{ and } \exists u_1 \in \mathcal{U}_1, u_2 \in \mathcal{U}_2 \text{ making } x^+ = \mathbf{A}x + B_1 u_1 + B_2 u_2 + d \in \Omega, \text{ for } \forall d \in \Phi.
\]

Such a set can be computed by recursively applying potryagin difference and projection [12]. In details, the computation is provided in Algorithm 1.

Algorithm 1 Computation of the invariant set

\begin{itemize}
\item Initialization: \(\Omega_0 = \mathcal{X} \cap \mathcal{X}_u, k = 1 \text{ and } flag = 0\)
\item while \(\Omega_k \text{ is not empty } \& k < N_{\text{max}} \text{ do}
\item \hspace{1em} Step 1: Take the pontryagin difference towards \(\Omega_0\)
\item \hspace{2em} \(\Omega_1 = \Omega_0 - \Phi\)
\item \hspace{1em} Step 2: Construct a set \(\Omega_2\) on \([x^T, u_1^T, u_2^T]^T\)
\item \hspace{2em} \(\Omega_2 = \{[x^T, u_1^T, u_2^T]^T : [A B_1 B_2][x^T, u_1^T, u_2^T]^T \in \Omega_1, \}
\item \hspace{2em} \(x \in \Omega_0, u_1 \in \mathcal{U}_1, u_2 \in \mathcal{U}_2\}\}
\item \hspace{1em} Step 3: Take projection on \(x\)
\item \hspace{2em} \(\Omega_3 = \text{Proj}_x(\Omega_2)\)
\item \hspace{2em} \(= \{x : \exists u_1 \in \mathcal{U}_1, u_2 \in \mathcal{U}_2, A x + B_1 u_1 + B_2 u_2 \in \Omega_1, x \in \Omega_0\}\}
\item \hspace{2em} \(k = k + 1\)
\item \hspace{2em} if \(\Omega_3 = \Omega_0\) then
\item \hspace{3em} Display 'The invariant set is obtained.'
\item \hspace{3em} Set \(\Omega = \Omega_0\).
\item \hspace{3em} \(flag = 1\)
\item \hspace{1em} break
\item \hspace{1em} \end{itemize}
\item end if
\item end while
\item if \(flag = 0\) then
\item Display 'The invariant set does not exist.'
\item end if
\end{itemize}

If Algorithm 1 can stop within finite iterations with \(\Omega_0\) non-empty, we can take \(\Omega = \Omega_0\) and incorporate the invariant set into the model predictive controller design as

\[
\min_{\mathbf{U}_p(t)} J(t)
\]
\[
s.t. (4), (5)
\]
\[
x_p(t+i) \in \mathcal{X}_u \\
u_1(t+i-1) \in \mathcal{U}_1 \\
u_2(t+i-1) \in \mathcal{U}_2 \\
x_p(t+i) \in \mathcal{X}, i = 1, 2, \ldots, p
\]
\[
\mathbf{A}x(t) + B_1 u_1(t) + B_2 u_2(t) \in \Omega - \Phi.
\]

Theorem 1: The constraint in (8) guarantees that if \(x(t) \in \Omega\), then the optimization in (7) is feasible at time \(t\). By implementing the controller designed in (7), it is ensured that \(x(t+1) \in \Omega\).

Proof: The theorem can be proved in two steps: (1) First step: if \(x(t) \in \Omega\), then the optimization problem in (7) is feasible at time \(t\).
$x(t) \in \Omega$ implies there exists a $u_1(t) \in U_1$ and a $u_2(t) \in U_2$ making $Ax(t) + B_1u_1(t) + B_2u_2(t) + d(t) \in \Omega$ for any $d \in \Phi$. According to the definition of pontryagin difference, this is equivalent to

$$Ax(t) + B_1u_1(t) + B_2u_2(t) \in \Omega - \Phi.$$ 

Thus, the constraint in (8) can be satisfied. Satisfaction of other constraints can be easily proved since $\Omega$ is subset of both $X$ and $X_u$.

(2) Second step: if the constraint in (8) is guaranteed, the actual state at $t + 1$ satisfies

$$x(t + 1) \in \Omega.$$ 

The actual state $x$ at $t + 1$ satisfies

$$x(t + 1) = Ax(t) + B_1u_1(t) + B_2u_2(t) + d(t),$$

according to the definition of pontryagin difference and (8), it is guaranteed that for any $d(t) \in \Phi$,

$$x(t + 1) \in \Omega.$$ 

The above two steps can be recursively applied, then the consistent feasibility of the optimization in (7) can be proved.

Moreover, feasibility of the optimization guarantees satisfaction of the control requirements.

Based on the optimization in (7), inputs from $t$ to $t + p_n - 1$ are computed. According to the receding horizon strategy, only the inputs corresponding to the $t$-th time step are applied, namely $u_1(t)$ and $u_2(t)$. Then, at $t + 1$, the optimization in (7) is re-conducted to derive the inputs.

IV. SIMULATION

In this section, we consider a barrel with four temperature zones in an extrusion process. The dynamic model is as

$$x(t + 1) = Ax(t) + B_1u_1(t) + B_2u_2(t) + d(t)$$

with

$$A = \begin{bmatrix} 0.965 & 0.01022 & 0 & 0 \\ 0.00982 & 0.968 & 0.01072 & 0 \\ 0 & 0.0105 & 0.963 & 0.017 \\ 0 & 0 & 0.01072 & 0.962 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 0.0217 & 0 & 0 & 0 \\ 0 & 0.0219 & 0 & 0 \\ 0 & 0 & 0.0213 & 0 \\ 0 & 0 & 0 & 0.0211 \end{bmatrix}$$

$$B_2 = \begin{bmatrix} 0.031 & 0 & 0 & 0 \\ 0 & 0.042 & 0 & 0 \\ 0 & 0 & 0.035 & 0 \\ 0 & 0 & 0 & 0.033 \end{bmatrix}.$$ 

Constraints on inputs and states are set as

$$0 \leq u_1^i(t) \leq 200$$

$$0 \leq u_2^i(t) \leq 200$$

$$x^i(t) \geq 0, \ i = 1, 2, 3, 4$$

with $u_1^i, u_2^i$ and $x^i$ denotes the $i$-th coordinate of the vector $u_1, u_2$ and $x$. The mechanical heat generated by shearing satisfies

$$0 \leq d^i \leq 10;$$

The safety constraint is set as

$$x^i(t) \leq 250.$$ 

Set point for each temperature zone is set as

$$x^1_r(t) = 200; \ x^2_r(t) = 210; \ x^3_r(t) = 220; \ x^4_r(t) = 230$$

with $t \in [1, 200]$. Assume the initial state for each zone is the same and taken as

$$x^1(0) = x^2(0) = x^3(0) = x^4(0) = 30.$$

- Case 1: Small mechanical heat

Firstly, assume the screw rotation is relatively slow, and therefore the heat generated by shearing is not too large. In this simulation, the heat generated is depicted in Fig 1. Then, based on the algorithm given in (7), a tracking controller can be designed. The tracking performance is shown in Fig. 2. Temperature in each zone can follow the set point. Inputs corresponding to the heaters and coolers are plotted in Fig. 3 and 4. Since the mechanical heat keeps to be small, the coolers are kept to be turned off. Only the heaters take actions to maintain the barrel temperature. In addition, due to the induction heat, the heaters fixed in the second and third zone are almost closed after $t = 50$, but the heaters in the first and forth zone keep to be active.

- Case 2: Excessive mechanical heat

In this part, we assume the screw rotation speeds up from $t = 100$ to $t = 150$. Therefore, the mechanical heat generated by shearing increases significantly as shown in Fig. 5. Fig. 6.
shows that the proposed method can ensure the temperature below the given safety bound even when the screw rotation speeds up (from \( t = 100 \) to \( t = 150 \)). Fig. 7 and Fig. 8 show that, the active controller switch to the coolers from the heaters when the heat \( d \) increases, and switch backward when \( d \) decreases. There is no chattering during this process. Moreover, the use of heaters and coolers is well separated. Heaters and coolers do not work simultaneously. Thus, energy can be saved. This example also shows that small variations of the mechanical heat may not cause switch between the two controllers. Only when large amount of excessive heat is generated, the switch occurs. In this way, frequent switches are also avoided.

V. CONCLUSIONS

In this paper, we designed a multi-objective model predictive control strategy for the control of barrel temperature in an extrusion process with both heaters and coolers. The proposed method can effectively avoid frequent switch between the two types of control inputs. In addition, the method can also prevent simultaneous activation of the two, which helps to reduce energy consumption.

ACKNOWLEDGMENT

The authors gratefully acknowledge the financial support from Guangdong Innovative and Entrepreneurial Research Team Program (No. 2013G076) and The National Natural Science Foundation of China project (No. 61433005).
REFERENCES


![Fig. 6. Case 2: Tracing performance: x.](image)

![Fig. 7. Case 2: Inputs corresponding to the heaters: u1.](image)

![Fig. 8. Case 2: Inputs corresponding to the coolers: u2.](image)