A performance optimization algorithm for controller reconfiguration handling actuator faults in fault tolerant distributed model predictive control

Alexey Zakharov
Aalto University, Department of Biotechnology and Chemical Technology, P.O. Box 16100, FI-00076 Aalto University, Finland. E-mail: alexey.zakharov@aalto.fi

Elena Zattoni
Alma Mater Studiorum University of Bologna, Department of Electrical, Electronic and Information Engineering “G. Marconi”, 40136 Bologna, Italy. E-mail: elena.zattoni@unibo.it

Miao Yu
Aalto University, Department of Biotechnology and Chemical Technology, P.O. Box 16100, FI-00076 Aalto University, Finland. E-mail: miao.yu@aalto.fi

Sirkka-Liisa Jämsä-Jounela
Aalto University, Department of Biotechnology and Chemical Technology, P.O. Box 16100, FI-00076 Aalto University, Finland. Tel: + 358 9 470 25826428, sirkka-liisa.jamsa-jounela@aalto.fi, http://chemtech.aalto.fi/en/

KEY WORDS Distributed model predictive control, fault tolerant control, controller reconfiguration, constrained optimization

ABSTRACT

This paper presents a performance optimization algorithm for controller reconfiguration in fault tolerant distributed model predictive control for large-scale systems. After the fault has been detected and diagnosed, several controller reconfigurations are proposed as candidate corrective actions for the fault compensation. The solution of a set of constrained optimization problems with different actuator and setpoint reconfigurations is derived by means of an original approach that exploits the information on the active constraints in the non-faulty subsystems, so as to split the global optimization problem into two optimization subproblems, which enables the on-line computation burden to be greatly reduced. Subsequently, the performances of different candidate controller reconfigurations are compared, and the better performing one is selected and then implemented to compensate the fault effects. Simulation results on a benzene alkylation process have demonstrated the efficacy of the proposed performance optimization algorithm in the fault tolerant distributed model predictive control.

1 INTRODUCTION

Increased global competition, higher product quality requirements and environmental regulations have forced the process industry to continuously optimize efficiency and profitability. Advanced control strategies, such as model predictive control (MPC), have made it possible to run processes close to the quality and safety limits thereby increasing profitability, whilst ensuring better end product quality and enhancing safety in plants. In the engineering practice, one centralized MPC usually cannot handle the whole large-scale process; instead, several MPCs may work together in a distributed manner to exchange the information of each system to achieve the control objectives. To this end, highly efficient distributed control methods have been developed over the past decades. For instance, Scheu and Marquardt /1/ have developed a distributed model predictive control (DMPC) methodology based on a distributed optimization algorithm, which relies on a coordination mechanism using the first-order sensitivities of the objective functions of neighboring systems. This proposed DMPC can effectively reduce the computation burden and overcome possible communication limitations of the centralized MPC.

Conventional control schemes are developed under the assumption that sensors and actuators are free from faults; however, the occurrence of faults causes degradation in the closed-loop performance and also has an impact on safety, productivity and plant economy. As a result, the research focus is shifting towards advanced management of abnormal situations, such as process disturbances and faults, which still provides great
possibilities for further improvement of the process efficiency. To this end, fault tolerant control (FTC) has attracted much attention in the area of engineering practice in recent years.

The corrective actions of FTC can be categorized into two types: fault accommodation and controller reconfiguration, whose difference lies in whether the controller setting will change for the compensation of fault effects. Thus, Pranatyasto and Qin /2/ studied the databased FTC with a simulated fluid catalytic cracking unit, where the sensor faults were detected by principal component analysis and accommodated in the MPC. On the other hand, Gani et al. /3/ developed two alternative single-input single-output controls for a polyethylene reactor, by manipulating different actuators: the temperature of a feed flow and a catalyst flow rate. In the case of an actuator failure, the control relying on the healthy actuator was applied. In large-scale systems, it is however difficult or impossible to develop back-up control strategies for all possible faults. Besides using redundant actuators, another approach for the controller reconfiguration is to define a new setpoint for the faulty system. The nominal process operating conditions can sometimes become infeasible because of the fault and in such a case, a new operating point must be defined. Thus, Chilin et al. /4/ proposed the use of the feasible steady state closest to the nominal steady state of the system as the new target operating points.

Currently, most of the FTC systems in the literature are based on a centralized MPC for the whole process. In order to bridge the gap between FTC and DMPC for large-scale systems, a framework for the design of a fault tolerant distributed model predictive control (FTDMPC) strategy is presented herein. After the fault has been detected and diagnosed and several candidate reconfigured controls have been proposed, the performance optimization algorithm, which means predicting the trajectories of process variables under distributed model predictive control, is performed. Then, the results are utilized to check whether the candidate reconfigured controls are able to drive the system to the new operating conditions and to evaluate the performance during the transition period. Thus, the most suitable candidate reconfigured controller is selected and its feasibility is ensured.

The remainder of this paper is organized as follows. In Section 2, a general idea of FTDMPC is introduced, which focuses on the function of the performance optimization algorithm for the controller reconfiguration. Section 3 shows how the formulation of the original DMPC is modified in the presence of a fault and how to reduce the computational burden implied by its solution with the introduction of suitable, motivated assumptions. In Section 4, simulation results from a benzene alkylation process are provided to demonstrate the effectiveness of the devised approach. Finally, Section 5 outlines the overall conclusions.

2 OUTLINE OF THE FAULT TOLERANT DISTRIBUTED MPC

The fault tolerant distributed model predictive control scheme for large-scale systems devised in this work mainly includes the following elements: distributed model predictive control, fault detection and diagnosis (FDD), controller reconfiguration based on the performance optimization algorithm. The overall structure of FTDMPC is shown in Figure 1.

![Figure 1. Outline of FTDMPC for large-scale system](image-url)
When the fault is detected and diagnosed, some possible reconfigured controllers can be designed to achieve the control aims in the presence of the faults. For instance, in case of an actuator fault (such as actuator blocking), the faulty actuator is usually replaced by alternative actuators or some actuator constraints are modified, e.g., according to degradation of the faulty actuator capacity. Hence, several reconfigured control settings can be generated for further evaluation.

In case that the current operating condition is infeasible for the faulty plant, a new operating condition needs to be defined according to the control objectives with the fault information provided by the FDD element. The new operating conditions can be selected from the set of steady states of the system under faulty dynamics. As an additional constraint, the target operating conditions in the downstream units must be disturbed as little as possible, Gandhi and Mhaskar /5/. In particular, one of the requirements is to maintain the set of current active constraints relating to the non-faulty systems as they were at the nominal operating conditions.

With each possible actuator and setpoint reconfiguration, the MPC turns out to be a different constrained optimization problem. The performance optimization algorithm aims at selecting one of the ensuing corrective actions by evaluating performance of each reconfigured controller action. With the condition that the active constraints in the non-faulty subsystems remain the same as they were in the nominal conditions, the global optimization of MPC can be split into two nested subproblems. As a result, the on-line computation burden is reduced and the consequent selection of the better performing controller can be achieved before the system state is driven far away from the nominal operating conditions.

### 3 THE IDEA OF THE PERFORMANCE OPTIMIZATION ALGORITHM

The large-scale system consists of the interconnection of a set of $N$ discrete-time linear time invariant systems described by:

$$\begin{align*}
    x_i(k+1) &= \sum_{j=1}^{N} A_{ij} x_j(k) + B_i u_i(k), \\
    y_i(k) &= C_i x_i(k)
\end{align*}$$

(1)

where $k$ is the time variable, $x_i$ is the state, $u_i$ is the control input and $y_i$ is the output. It is assumed that the DMPC objective is a quadratic function of the control inputs and the outputs, and there are sets of inequality constraints associated to every subsystem. The symbols $u_i$ and $y_i$ respectively denote the vectors collecting the sequences of the control inputs and the corresponding outputs over the prediction time interval $k_p$, for the given initial states $\xi$:

$$\begin{align*}
    u_i &= \text{vect} \{ u_i(0), u_i(1), \ldots, u_i(k_p - 1) \}, \\
    y_i &= \text{vect} \{ y_i(1), y_i(2), \ldots, y_i(k_p) \}, \\
    u^* &= \text{vect} \{ u_1, u_2, \ldots, u_N \},
\end{align*}$$

(2) (3) (4)

Note that, by means of simple algebraic manipulations, the DMPC objective can be written as

$$J = u^T \Psi u^* + \phi^T u^* + \rho,$$

(5)

where $\Psi$, $\phi$ and $\rho$ are matrices of appropriate dimensions.

When a fault is diagnosed in subsystem $i$, the large-scale system inputs $u^*$ are rearranged into

$$\begin{bmatrix} u^*_h \\ u^*_f \end{bmatrix},$$

(6)

where terms $u^*_h$ and $u^*_f$ represent the inputs of the healthy subsystems and the faulty subsystem respectively.

Assuming of taking into account only active constraints in non-faulty systems:

$$F_h u_h + F_f u_f + E \xi + d = 0,$$

(7)

where $\xi$ contains the states of the subsystems, at the first stage objective (5) can be minimized explicitly with respect to $u^*_h$. Thus the inputs of non-faulty subsystems $u^*_h$ can be expressed linearly through the inputs of the faulty sub-system $u^*_f$ and the current plant state $\xi$. The second stage of the optimization involves substituting the obtained expression to the objective (5) and its optimization with respect to $u^*_f$ in presence of inequality constraints associated to the faulty subsystem. Next, the obtained values of $u^*_f$ and $u^*_h$ can be used to predict the plant state one time step ahead, and the procedure can be repeated to obtain the prediction of the closed loop system performance in presence of the diagnosed fault.
4 SIMULATION RESULTS

This section illustrates the main results obtained by testing the fault tolerant distributed model predictive control scheme devised in this work on the benzene alkylation process. First, the benchmark process is described briefly, then, the DMPC and FDD methods are introduced separately. Finally, the emphasis is given to FTC, especially with regard to the implementation of the performance optimization algorithm.

4.1 Process description

The alkylation of benzene, shown in Figure 2, is a benchmark process which has been extensively studied with DMPC /1, 4/. The plant consists of five units: i.e., four continuous stirred-tank reactors (CSTR) and one flash separator. The purpose of the plant is to produce ethylbenzene (C) by the reaction of the raw materials benzene (A) and ethene (B). Benzene and ethene are fed into the cascaded CSTR 1, 2, and 3, where ethylbenzene is produced. In addition, the by-product diethylbenzene (D) is produced. The stream $F_7$ is fed into the flash separator, where unreacted benzene is separated from the product. The vapor stream is recycled; one part goes directly to CSTR 1, the other to CSTR 4. There, additional diethylbenzene (D) is fed and a transalkylation process leads to the reaction of benzene and diethylbenzene into ethylbenzene. The effluent of CSTR 4 is fed to the flash separator.

The mathematical model consists of material balances for each component and an energy balance for each unit of the plant, which results in a system model that includes a total of 25 states. The states of the process consist of the concentrations of A, B, C and D in each of the five tanks and the temperatures of the tanks. The state is assumed to be available continuously to the controllers. In addition, the model includes nonlinear reaction kinetics as well as a nonlinear description of the phase equilibrium in the flash separator, leading to a total of approximately 100 equations. Each of the tanks has an external heat/coolant input. In the normal condition, the manipulated inputs to the process are the heat injected to or removed from the five tanks, $Q_1$, $Q_2$, $Q_3$, $Q_4$ and $Q_5$ ($u_1$, $u_2$, $u_3$, $u_4$ and $u_5$, respectively). The feed stream flow rates to CSTR 2 and CSTR 3, $F_4$ and $F_6$, are the backup manipulated variables ($u_6$ and $u_7$) which are activated for the controller reconfiguration when a fault is detected. The steady-state inputs, $u_{i^{*}}$, $i = 1, \ldots, 7$, as well as the steady-state temperatures in the five tanks are shown in Table 1. The nonlinear model is linearized (by a finite-difference approach) at this operating point.

![Figure 2. Process flow diagram for alkylation of benzene /1/](image)

4.2 Distributed MPC strategy

In this work, the sensitivity-driven DMPC in /1/ is utilized as the base controller for the alkylation of benzene process. The whole system is divided into two groups, one includes CSTR 1, CSTR 2 and CSTR 3, the other
contains CSTR 4 and the flash separator. Thus, the process is under the control of two local controllers, and information is exchanged between them. In the non-faulty situation, only inputs $u_1, u_2, u_3, u_4$ and $u_5$ are actuated, which means the first distributed controller (DMPC 1) controls the values of $Q_1, Q_2$ and $Q_3$, and the second distributed controller (DMPC 2) controls the values of $Q_4$ and $Q_5$. When the inputs $u_6$ and $u_7$ are actuated in the faulty situation, they are used to replace the corresponding faulty actuators. The inputs $\Delta u_i(t)$ are discretized as a piecewise constant with sampling time $\Delta t = 10s$. The control horizon is $N = 5$, and the prediction horizon is $P = 20$. In addition to the input constraints, the temperatures in the tanks were also bounded to keep the process conditions close to the nominal point.

<table>
<thead>
<tr>
<th>Table 1. Steady-State Inputs and Temperatures</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_{1s} = -2.0 \times 10^6$ J/s</td>
</tr>
<tr>
<td>$u_{2s} = -2.0 \times 10^6$ J/s</td>
</tr>
<tr>
<td>$u_{3s} = -2.0 \times 10^6$ J/s</td>
</tr>
<tr>
<td>$u_{4s} = 4.1 \times 10^6$ J/s</td>
</tr>
<tr>
<td>$u_{5s} = -0.01 \times 10^6$ J/s</td>
</tr>
<tr>
<td>$u_{6s} = 8.697 \times 10^{-4}$ m$^3$/s</td>
</tr>
</tbody>
</table>

### 4.3 Case study 1

We consider an actuator fault occurs at $t = 300s$ (sample 30): $u_2$ is blocked at 95% of its steady-state value, that is, $u_2 = -1.9 \times 10^6$ J/s. Obviously, the temperature in CSTR 2 will be increasing from that time if no FTC is implemented. Since the current operating point is not feasible with the existing actuators, one possible solution is to activate another actuator in order to compensate for the efficiency loss in $u_2$. To demonstrate the function of the performance optimization algorithm for controller reconfiguration, two back-up actuator reconfigurations are investigated. The first is to activate the feed stream flow rates to CSTR 2, $u_6$, and the second is to activate the feed stream flow rates to CSTR 3, $u_7$. Figure 3 depicts the test result with activating $u_6$ and $u_7$ under the current operating point. From the trajectories under performance optimization algorithm, it is clear to see that the temperatures can be driven to setpoint after 12 steps under the effect of $u_6$. In the second case, the trajectories under performance optimization algorithm demonstrate irrefutably that activating $u_7$ does not help to compensate the fault. After the comparison, it can be decided to implement the first control reconfiguration at $t = 310s$ (sample 31), which will result in the temperatures converging to the setpoint with reconfigured controller at $t = 430s$ (sample 43).

![Figure 3](image.png)
4.4 Case study 2

In this part, a case study where the current operating point is not feasible with either original control strategy or any reconfigured actuators is outlined. Hence, another operating point must be designed based on the characteristics of the fault. We consider an actuator fault occurs at \( t = 300 \) s (sample 30): \( u_1 \) is blocked at 97.5% of its steady-state value, that is, \( u_1 = -1.95 \times 10^6 \) J/s and obviously, the temperature in CSTR 1 will be increasing from that time. It is clear that the fault in \( u_1 \) in CSTR 1 cannot be compensated by current control strategy or activating \( u_6 \) and \( u_7 \) in CSTR 2 and 3. This was confirmed by testing the trajectories under performance optimization algorithm under three different control configurations: the current one and the reconfigurations activating \( u_6 \) and \( u_7 \) respectively. Thus, one possible solution is to increase the setpoint for \( T_1 \). Another choice is to decrease the setpoint for the temperature in the flash separator, \( T_8 \). Since the recycled vapor stream goes from flash separator to CSTR 1, the cooling of this stream can also lead to the decreasing of \( T_1 \). The new operating point is designed as follows:

\[
T_{1s} = 473.36K, T_{2s} = 472.35K, T_{3s} = 472.39K, T_{4s} = 471.00K, T_{5s} = 473.00K.
\]

Figure 4 shows the trajectories under performance optimization algorithm for the future 20 steps with newly designed setpoint at time \( t = 320 \) s (sample 32). It can be clearly seen that both second and third controllers can obtain very good performance. After checking the difference between the predicted trajectory and setpoint, it was found that the third controller performs slightly better than the second one and as a result, \( u_7 \) is activated at time \( t = 320 \) s (sample 32).

5 CONCLUSIONS

This paper presents a performance optimization algorithm for controller reconfiguration in FTDMPC of large-scale systems. The performance optimization algorithm aims to check the ability and performance of the candidate reconfigured controllers in driving the process variables to the newly defined operating conditions. Under the assumption that the active constraints in non-faulty systems remain the same as they are at the nominal operating conditions, the global DMPC is split into two subproblems, which achieves the objective of rendering the computational burden compatible for on-line processing. The efficacy of the proposed performance optimization algorithm for controller reconfiguration has been demonstrated with two case studies on the alkylation of benzene process.

6 REFERENCES