Proceedings of the 2006 American Control Conference
Minneapolis, Minnesota, USA, June 14-16, 2006

Application of Model Predictive Control for Fault Tolerant System Using Dynamic Safety Margin

M. Abdel-Geliel, E. Badreddin, A. Gambier, *Member, IEEE*

Automation Lab, University of Mannheim, Germany
elgeliel@ti.uni-mannheim.de, *badreddin@ti.uni-mannheim.de*, *gambier@ti.uni-mannheim.de*

Abstract—**Model Predictive Control (MPC) has the ability to cope with hard constraints on control and state. It has, therefore, been widely applied in most industries specially, petrochemical industries. Dynamic Safety Margin (DSM) is a performance index used to measure the distance between a predefined safety boundary, described by a set of inequality constraints, in state space and system trajectory as it evolves. Designing MPC based on DSM is especially important for safety critical system to maintain a predefined margin of safety during transient and steady state. In this work, MPC based on DSM is used in fault tolerant control (FTC) design. The proposed method of FTC is suitable for single and multi-model system according to the fault type and fault information. It can compensate missed information about the fault and uncertainties in the faulty model.**

I.INTRODUCTION

A Fault tolerant control (FTC) system is a control system that can accommodate system component faults and is able to maintain stability and acceptable degree of performance not only when the system is fault-free but also when there are component malfunctions [5]. FTC prevents faults in a subsystem from developing into failures at system level. FTC system design techniques can be classified as passive and active (PFTC and AFTC) [4]-[6]. In PFTC, a system may tolerate only a limited number of faults, which are assumed to be known prior to the design of the controller. Once the controller is designed, it can compensate for anticipated faults without any access of on-line fault information. PFTC system treats the faults as if they were sources of modeling uncertainty [5]. AFTC either compensates the effect of faults by selecting a pre-computed control law, or by synthesizing a new control law in real-time. Both methods need a fault detection and identification (FDI) algorithm to identify the fault-induced changes and to reconfigure the control law on-line.

Model Predictive control (MPC) is an optimization based strategy that uses a plant model to predict the effect of potential control action on the evolving state of the plant. At each time step, an optimal control problem is solved and the first input vector is injected into the plant until a new measurement is available. The updated plant information is used to formulate and solve a new optimal control problem [7]-[10]. Since MPC is formulated as an optimization problem, inequality constraints are a natural addition to the controller [10]. The ability to handle explicitly hard constraints on control and states may be viewed as one of the major factors of the success of MPC in process control. Therefore, it has been widely applied in petrochemical and related industries. Hence, application of MPC in FTC is very important and useful, where most of the processes have control and state constraints, which specify the actuator limits and safety requirements of the components. Although, constraints improve the appeal of MPC as advanced control strategy, they make difficult the controller implementation.

The idea of using MPC in FTC is firstly discussed in [11] and implemented on a simulation model of EL AL Flight 1862 in [12]. Both references argue that MPC provides suitable implementation architecture for fault tolerant control. The representation of both faults and control objective is relatively natural and straightforward in MPC. Some faults can be represented by modifying the constraints in MPC problem definition. Other fault can be represented by modifying the internal model used by MPC [12], [9]. In addition, MPC has a good degree of fault tolerant to some faults, especially actuator faults, under a certain conditions, even if the faults are not detected (PFTC).

According to the definition of DSM [1]-[3], it indicates how far the system state is from a specified safety region, which determined by a set of inequality constraints. It is known that, the information about the fault from FDI in most cases is not very accurate or sufficient. Moreover, the uncertainties exist in faulty model. Therefore, considering DSM constraints in recovery controller, especially MPC, is useful to compensate the unavailable fault information and model uncertainties ([23]). In addition, DSM index can help in adapting MPC controller in order to find a feasible solution for constrained MPC and to satisfy the acceptable degraded performance. Thus, designing an FTC system against system faults to achieve an acceptable degraded of performance without violating the safety requirements of the overall system is the focus of the work presented here.

The paper is organized as follows: Dynamic safety margin and safety controller requirements are defined in section II. It is followed by the discussion about MPC with constraints and the implementation of DSM in MPC in Section III. The proposed FTC system based on MPC and DSM is explained in Section IV. An implementation example is illustrated in Section V. Finally, conclusion and future work are given in Section VI.

II. DMS DEFINITION AND SAFETY CONTROL

The idea of DSM is introduced in [1], [2]. Here, the general idea will briefly be explained. Let **X** be the state space in $\Re^n$, and consider that a subspace $\mathbf{\Phi} \subseteq \mathbf{X}$, which defines the safe operation region for some system state variables $\boldsymbol{x} \in \Re^m$ in the state subspace $\mathbf{\Phi}$, can be specified by a set of inequalities $\{\phi_i(\mathbf{x}) \leq 0 \,|i=1,...,q\}$, where ϕ_i: $\Re^m \rightarrow \Re$. $\phi_i(\mathbf{x}) > 0$ indicates unsafe operation (Fig. 1). It is assumed that the system is stable in the sense of Lyapunov and that the safe region is fully contained in the stability region. Starting with the initial condition $\mathbf{x}_0$, the system trajectory will evolve to the operating point $\mathbf{x}_s$ traversing the state space

1-4244-0210-7/06/$20.00 ©2006 IEEE

with varying distance to the safety boundary. DSM is defined in this case as the instantaneous shortest distance $\delta(t)$, between the system state vector of interest and a predefined boundary $\phi(\mathbf{x}) = 0$ in this subspace of state variables. At the operating point, $d\delta(t)/dt = 0$ and $\delta(\cdot)$ reaches a constant value indicating the Stationary Safety Margin (SSM). Most industrial designs are made to satisfy SSM of specified values.

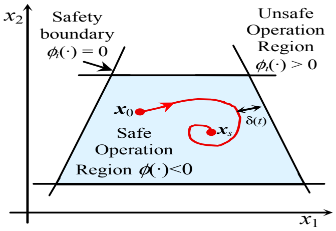

x_2
Safety boundary $\phi_i(\cdot) = 0$
Unsafe Operation Region $\phi_i(\cdot) > 0$
$\boldsymbol{x}_0$
$\boldsymbol{x}_s$
$\delta(t)$
Safe Operation Region $\phi(\cdot)<0$
x_1

Fig. 1: DSM definition

In general, the safe-operation region $\mathbf{\Phi}$ is defined by a set of inequalities given by

$$\mathbf{\Phi} = \{\phi_i(\mathbf{x}) \leq 0 | i = 1,\dots,q\} \tag{1}$$

and DSM is defined as

$$\delta(t) = \min_{1 \leq i \leq q} \delta_i(t) \tag{2}$$

$$\delta_i(t) = s(t) \cdot \left\| \mathbf{x}_i |_{\phi_i(\mathbf{x}_i)=0} - \mathbf{x} \right\|_{\min}, \ \mathbf{x} = [x_1, x_2, \dots, x_m]^T \tag{3}$$

where $s(t) = \begin{cases} +1 & \text{if } \boldsymbol{x} \text{ inside the safe operation region} \\ -1 & \text{if } \boldsymbol{x} \text{ outside the safe operation region} \end{cases}$

and $\| \cdot \|_{\min} \triangleq$ shortest distance from $\boldsymbol{x}(t)$ to ϕ. For all boundaries, the distance vector

$$\boldsymbol{d}(t) = [\delta_1(t),\ \delta_2(t), \cdots,\ \delta_q(t)]^{\mathrm{T}}\ . \tag{4}$$

Variable q is the number of defined inequalities and m the number of state variables relevant to safety. Notice that $m \leq n$, where n is the dimension of the state-space.

Safety is one of the important specifications of the controlled system. Safety control problem requires moving the system state from a given set of initial states in its state space to a predetermined safe region achieving, in addition, the desired nominal performance of the system. Thus, the controller should move the state in order to satisfy $\delta(\cdot) \geq 0$. A positive value of DSM, i.e. $\mathbf{d}(\cdot) \geq 0$, means the nominal performance is achieved. Contrarily, a negative value of DSM $\delta(\cdot) \leq 0$ means that the system has been exposed to a large disturbance and/or model uncertainties or a fault exists.

To maintain the system states within a predefined margin of safety, the value of DSM must be considered in controller design. Implementing DSM in a controller can be achieved by various methods. The safety region Φ may be considered as a controlled invariant set [24] if there is a controller that assure DSM positive for the closed loop system.

Definition 1: The set $\Phi \subset \Re^{n}$ is said (robustly) controlled invariant for the system

$$\begin{aligned} \Delta\mathbf{x}(t) &= f(\mathbf{x}(t), \mathbf{u}(t), \mathbf{w}(t)), \\ \mathbf{y}(t) &= g(\mathbf{x}(t)) \end{aligned} \tag{5}$$

If for all $\mathbf{x}(0) \in \Phi$ there is a continuous feedback control law

$$\mathbf{u}(t) = \psi(\mathbf{y}(t)) \text{ or } \mathbf{u}(t) = \psi(\mathbf{x}(t)) \tag{6}$$

which assures the existence and uniqueness of the solution, $\mathbf{x}(t) \in \Phi$, and then Φ is positively invariant for the closed loop system. where $\mathrm{x}(t) \in \Re^n$ is the system state, $\mathrm{u}(t) \in \Re^m$ is the control input, $\mathrm{y}(t) \in \Re^p$ is the output, $\mathrm{w}(t) \in \mathrm{W} \subset \Re^q$ is the external input (disturbance), W is assigned compact set, and Δ is the derivative operator in continuous time and shift operator in discrete time case.

Hence, the invariance condition can be defined as

$$\mathrm{dist}(\mathbf{x}, \Phi) = \inf_{\mathbf{x}_i \in \Phi} \left\| \mathbf{x} - \mathbf{x}_i \right\| = 0\,, \tag{7}$$

this means that DSM ≥ 0.

It is not possible in all cases to find a linear controller to a controlled invariant polytope, it is often necessary consider non-linear control laws [24].

In some cases, it is difficult to satisfy the safety, $\delta(\cdot) \geq 0$ and the desired nominal performance specially, if there is a fault. Therefore, it is necessary to accept some degree of performance degradation after occurrence of a fault in order to satisfy the safety. Some ideas for implementing DSM in controller design are stated in [1], [3]. In those contributions, adapting PID controller parameters, switching controller, Fuzzy controller and/or optimal control are highlighted.

III. CONSTRAINED MODEL PREDICTIVE CONTROL

Model predictive Control (MPC) or receding horizon control (RHC) is a form of control in which the current control action is obtained by solving on-line, at each sampling instance, a finite horizon optimal control problem, using the current state of the plant as the initial state. The internal model is used to obtain prediction of system behavior over the finite horizon [7]-[10]. The optimization yields an optimal control sequence but only the first control of the sequence is applied to the plant and in the next sampling time, the complete calculation is repeated (*receding horizon principle*). This is the main difference from conventional control, which uses a pre-computed control law. It naturally handles the control of multivariable plant and takes account the information on constraints arising from equipment limitations, safety requirements, etc. In its usual form, it does this by combining linear dynamic models with linear inequalities, which seems to be very powerful combination, since the linear model keeps the dynamic simple, while the inequalities can be used to represent important nonlinearities, as well as constraints. The usual formulation of MPC using a quadratic cost functional combined with a linear model and linear inequalities leads to a quadratic programming optimization problem. The ability to handle explicitly hard constraints on control and states may be viewed as one of the major factors of the success of MPC in process control. It has, therefore, been widely applied in process industries. The control law of predictive controller, for a system defined by the state-space model, is determined from the minimization of a 2-norm measure of predicted performance [5]

$$J=\left(\sum_{i=N_1}^{N}\left\|\hat{\mathbf{e}}(i+k|k)\right\|_{\mathbf{Q}_i}^2+\sum_{i=0}^{N_u-1}\left\|\mathbf{u}(k+1)\right\|_{\mathbf{R}_i}^2\right) \tag{8}$$

Subject to

$$\begin{aligned}&\mathbf{x}(k+1)=\mathbf{A}x(k)+\mathbf{B}u(k)\\&\mathbf{y}(k)=\mathbf{C}\mathbf{x}(k)+\mathbf{D}\mathbf{u}(k)\\&\mathbf{x}(k+i|k)\in\mathbf{X}_s\\&\mathbf{u}_{\min}\le\mathbf{u}(k+1)\le\mathbf{u}_{\max}\end{aligned} \tag{9}$$

where

$$\underset{\rightarrow}{\mathbf{u}}=\left[\mathbf{u}(k)\ \mathbf{u}(k+1)\cdots\mathbf{u}(k+N_u-1)\right]^{\mathrm{T}}\in\Re^{r.N_u};$$

$$\hat{\mathbf{e}}(k+1|k)=\mathbf{y}_d(k+i)-\hat{\mathbf{y}}(k+i|k)$$

$\|\boldsymbol{e}\|_Q^2=\boldsymbol{e}^{\mathrm{T}}\boldsymbol{Qe}$, $\hat{\mathbf{e}}(k+1|k)\in\Re^m$ is the predicted error between the desired and predicted response. $\boldsymbol{x}\in\Re^{\mathrm{n}}$ is the system state vector, $\mathbf{y}_{\mathrm{d}}\in\Re^m$ is the reference output vector. $\hat{\mathbf{x}}(k+1)$ is the prediction of $\mathbf{x}(k+i)$ made at instance k, $\mathbf{X}_s\subseteq\mathbf{X}$ is the set of state vectors satisfy all state constraints. **A**, **B**, **C** and **D** system parameter matrix of adequate dimensions, $\mathbf{Q}_i$ are the error weighting matrices, $\mathbf{R}_i$ are the input weighting matrices. N, N_1 and N_u are the maximum, minimum and control horizons, respectively.

Designing MPC based on DSM can be handled by replacing $\mathbf{X}_s$ by the safety region $\mathbf{\Phi}$. The state constraints can be written in the form

$$\hat{\delta}(k+1|k)\ \text{or}\ \hat{\mathbf{d}}(k+1|k)\ge 0 \tag{10}$$

where $\hat{\delta}(k+1|k)\in\Re$ and $\hat{\mathbf{d}}(k+1|k)\in\Re^q$ are the prediction of DSM and distance vector, between the predicted state and the boundaries of $\mathbf{\Phi}$, made at instance k respectively.

In most cases, the safe operation region can be defined by a set of linear inequalities. In case that the boundary function is nonlinear, it can be subdivided into two or more linear constraints (piecewise linear approximation). Hence, the boundary of $\mathbf{\Phi}$ can be formulated in general as a set of linear inequalities, $\{\phi_i(\boldsymbol{x})\le 0\,|i=1,...,q\}$;

$$\phi_i(\mathbf{x})=\boldsymbol{a}_i^{\mathrm{T}}\mathbf{x}-\mathrm{c}_i\le 0 \tag{11}$$

where $\boldsymbol{a}_i^{\mathrm{T}}\in\Re^n$, $\mathrm{c}_i\in\Re$ and $\{\mathbf{x}_i\,|\phi_i(\mathbf{x}_i)=0\}\subset\mathbf{\Phi}$is a subset of state vector where $\boldsymbol{a}_{\mathrm{i}}^{\mathrm{T}}\mathbf{x}_i=\mathrm{c}_i$. Thus, for the state vector **x**, $\delta(\cdot)$ can calculated [2] as

$$\delta_i(t)=\frac{c_i-\mathbf{a}_i\mathbf{x}(t)}{\|\mathbf{a}_i\|_2}\begin{cases}\ge 0\ \textbf{iff}\ \ \phi(\mathbf{x})\le 0\\<0\ \text{iff}\ \ \phi(\mathbf{x})>0\end{cases} \tag{12}$$

for all boundaries, the distance vector

$$\mathbf{d}(k)=[\delta_1(k)\quad\delta_2(k)\cdots\delta_q(k)]^{\mathrm{T}}$$

can be obtained from

$$\mathbf{d}(k)=\mathbf{d}_c-\mathbf{D}_a x(k)\in\Re^q \tag{13}$$

where $\mathbf{d}_c=\mathbf{D}_{ia}\,\mathbf{c}_c\in\Re^q$, $\mathbf{D}_a=\mathbf{D}_{ia}\,\mathbf{A}_c\in\Re^{q\times n}$ with

$$\boldsymbol{D}_{ia}=\mathrm{diag}\left(\frac{1}{\|\boldsymbol{a}_1\|_2},\frac{1}{\|\boldsymbol{a}_2\|_2},\cdots,\frac{1}{\|\boldsymbol{a}_q\|_2}\right)\in\Re^{q\times q}\ \text{and}$$

$$\boldsymbol{c}_c=\begin{bmatrix}c_1 & c_2 & \cdots & c_q\end{bmatrix}\in\Re^q;\ \boldsymbol{A}_{\mathrm{c}}=\begin{bmatrix}\boldsymbol{a}_1 & \boldsymbol{a}_2 & \cdots & \boldsymbol{a}_q\end{bmatrix}\in\Re^{q\times n}.$$

DSM $\delta(\cdot)$ is the minimum element in $\mathbf{d}(\cdot)$ as in (2). Consider (9) and (13) then a general form of error prediction and distance vector $\boldsymbol{d}$ are

$$\underset{\rightarrow}{\mathbf{e}}=(\underset{\rightarrow}{\mathbf{y}}-\mathbf{C}_a\mathbf{x}(k)+\mathbf{C}_B\underset{\rightarrow}{\mathbf{u}}) \tag{14}$$

$$\left.\begin{aligned}&\underset{\rightarrow}{\mathbf{d}}=\mathbf{d}_t+\mathbf{D}_a\mathbf{x}(k)-\mathbf{D}_b\mathbf{u}\ge 0\\&\quad\mathbf{u}_{\min}\le\mathbf{u}(k+1)\le\mathbf{u}_{\max}\end{aligned}\right\} \tag{15}$$

where

$$\underset{\rightarrow}{\boldsymbol{y}}=\begin{bmatrix}\boldsymbol{y}_d(k+N_1)\\\boldsymbol{y}_d(k+N_1+1)\\\vdots\\\boldsymbol{y}_d(k+N)\end{bmatrix}\in\Re^{m(N-N_1+1)},\ \underset{\rightarrow}{\boldsymbol{e}}=\begin{bmatrix}\boldsymbol{e}(k+N_1)\\\boldsymbol{e}(k+N+1)\\\vdots\\\boldsymbol{e}(k+N)\end{bmatrix}\in\Re^{m(N-N_1+1)},\ \underset{\rightarrow}{\boldsymbol{d}}=\begin{bmatrix}\boldsymbol{d}(k+1)\\\boldsymbol{d}(k+2)\\\vdots\\\boldsymbol{d}(k+N)\end{bmatrix}\in\Re^{q(N-N_1+1)}$$

$\mathbf{D}_a$, $\mathbf{D}_b$, $\mathbf{d}_t$, $\mathbf{C}_a$ and $\mathbf{C}_b$ are easily obtained from (12) and (13). The objective function according to (11) is

$$J=\underset{\rightarrow}{\mathbf{u}}^T\mathbf{M}\underset{\rightarrow}{\mathbf{u}}+2\mathbf{H}\underset{\rightarrow}{\mathbf{u}}+c_r \tag{16}$$

subject to (12), where

$$\mathbf{M}=\mathbf{C}_B{}^T\mathbf{Q}_t\mathbf{C}_B+\mathbf{R}_t;$$

$$\mathbf{H}=(\underset{\rightarrow}{\mathbf{y}}-\mathbf{C}_a\mathbf{x}(k))^T\mathbf{Q}_t\mathbf{C}_B;$$

$$\mathbf{c}_r=(\underset{\rightarrow}{\mathbf{y}}-\mathbf{C}_a\mathbf{x}(k))^T\mathbf{Q}_t(\underset{\rightarrow}{\mathbf{y}}-\mathbf{C}_a\mathbf{x}(k));$$

$$\mathbf{Q}_t=\begin{bmatrix}\mathbf{Q}_{N1}0\cdots 0\\0\,\mathbf{Q}_{N_1+1}\cdots\\\vdots\quad\ddots\\0\,0\cdots\mathbf{Q}_N\end{bmatrix}\in\Re^{m.(N-N_1+1)\times m.(N-N_1+1)},\ \mathbf{R}_t=\begin{bmatrix}\mathbf{R}_1 0\cdots 0\\0\,\mathbf{R}_2\cdots\\\vdots\quad\ddots\\0\,0\cdots\mathbf{R}_{N_u}\end{bmatrix}\in\Re^{r.N_u\times r.N_u}$$

Equation (16) is known as a multi-parametric quadratic programming (mp-QP) problem, for which there are some algorithms and methods to find a feasible solution, see for example [13]-[16]. On-line optimization of equation (13) gives the desired control sequence, which achieves the output performance and the safety performance. Note, $\boldsymbol{Q}$, $\boldsymbol{S}$, $\boldsymbol{R}$, N, N_1 and N_{u} are free design parameters.

If there is an uncertain input in the system, which can be considered as the additive fault information in our case, Min-Max predictive controller design can be used [18]-[19].

In the proposed algorithm, the DSM value is used to determine the priorities of the constraints in order to find a feasible solution. Moreover, it is used in adapting the objective function by changing the weighting matrices according to DSM in order to satisfy the accepted degraded performance and constraints.

Special case:

Assumed that, the safe operating region is convex, then the minimization of 2-norm of $\mathbf{d}(\cdot)$ moves the states to be inside the safe region system $\mathbf{\Phi}$. Hence, the 2-norm of $\mathbf{d}(\cdot)$ can be introduced as additional term in the main objective function (16). The objective function of the predictive controller in this case can be rewritten in the following form

$$J=\left(\sum_{i=N_1}^{N}\left\|\hat{\mathbf{e}}_d(i+k|k)\right\|_{\mathbf{Q}_i}^2+\sum_{i=0}^{N_u-1}\left\|\mathbf{u}(k+1)\right\|_{\mathbf{R}_i}^2\right) \tag{17}$$

Subject to (12),

where $\hat{\boldsymbol{e}}_d(\cdot)=[\hat{\boldsymbol{e}}^{\mathrm{T}}(\cdot)\hat{\boldsymbol{d}}^{\mathrm{T}}(\cdot)]^{\mathrm{T}}\in\Re^{m+q}$, $\overline{\boldsymbol{Q}}_i=\begin{bmatrix}\boldsymbol{Q}_i & 0\\0 & \boldsymbol{P}_i\end{bmatrix}$ and $\boldsymbol{P}_i=\boldsymbol{P}_{io}e^{-\delta(k)}$.

$\mathbf{P}_i$ is the weighting matrices for **d** and it is depend on the value of DSM (δ) i.e. if δ is negative then the weighting matrix increased and vice verse. **P** is a constant weighting matrix**.** The number of free design parameters in this case increased by $\mathbf{P}_0$.

The hard constraints, in this case, are restricted to control input. Solving problem in the form of (17) using direct optimization can be found in [8], [23]. The control problem (17) can also be solved using dynamic programming. The control law is given by the affine control law [17] but without integral action. Both methods are explained in more details in [23].

The solution of (17) is feasible if the control law satisfies the constraints of control signal $\mathbf{u}(i+k)$ otherwise, the objective function parameters should be adapted in order to satisfy the control constraints. Representing MPC with constraints in the form of (17) simplifies the solution and reduces the computation burden. Therefore, MPC with DSM will not be restricted for small process but it could be applied in large scale system.

IV. Fault Tolerance Based on MPC

MPC with constraints provides a suitable implementation architecture for fault tolerant control, as stated before. The degraded accepted performance is obtained using MPC by changing the objective function or using multi-objective function. According to the definition of DSM, MPC based on DSM constraints satisfy the safety requirements of the system and the accepted degraded performance. In addition, design based on DSM constraints can compensate the unavailable information about the faults where it is not easy to find an FDI system provides a full and exact information about the exist fault. Moreover, not all faults can be anticipated. Therefore, designing FTC based on DSM is useful to recover the system in case of anticipated faults to compensate the missed fault information. Moreover, controller based on DSM can maintain the safety operation with acceptable degraded performance in some cases of unanticipated faults.

Three controllers are configured in the proposed FTC design and used according to the following scenario:

- Under normal operating conditions, a nominal controller is designed to guarantee the system's stability and robust tracking performance in the presence of the modeling uncertainty.
- When a fault occurs, the nominal controller should guarantee the system signal boundary by checking DSM until the fault is detected.
- After fault detection (DSM is negative), the nominal controller is replaced by MPC controller based on DSM using the nominal system model to compensate for the effect of the (yet unknown) fault. This controller may recover some control performances (e.g., tracking of reference). This new controller should guarantee the boundedness of safety requirements and physical limitation of the components even in the presence of the fault.
- If the fault is isolated, then the MPC with DSM is reconfigured again using fault information by selecting the suitable faulty model to improve the control performances.

Since the reconfigured controller design is based on FDI system, the proposed FTC system is belong to active FTC systems. If FDI system failed to detect and isolate the fault, the proposed FTC will behave as PFTC, and the fault will be considered as a source of disturbance or uncertainties.

Remark 1: In case of PFTC, i.e. there is no FDI system, the controller is reconfigured based on the value of DSM. i.e., there are two controller configuration only nominal and MPC based on DSM using the nominal model (first FTC).

Remark 2: It is possible that, in some cases, the fault that has occurred cannot be isolated, for instance a fault whose functional structure is completely unknown a priori (i.e., does not belong to). Then, the first fault-tolerant controller guarantees some minimal performance (e.g., closed-loop stability). In this case, the second FTC cannot be activated.

In this approach, the fault is isolated using multi-model FDI scheme [2], [20]-[22], the reconfigurable controller is designed using MPC based on DSM and new faulty model selected according to FDI algorithm information. Fig. 2 shows general structure of proposed reconfigurable control scheme, which includes Blocks of reference models, MPC using safety region constraints, multi-model FDI system employing parameters and state estimation and supervisory control.

A. Multi-Reference Model and Command Control Block

To design a fault tolerant control (FTC) system, one of the most important issues to consider is whether to recover the original system performance/functionality completely or to accept some degree of performance degradation after occurrence of a fault [6], [15]. Most of the earlier work of FTC design concentrated on the philosophy to recover the default system performance as much as possible. In practice, however, because of a faulty part, the degree of the other system components capability could be significantly reduced. If the design objective still to maintain the original system performance, this will force the remaining parts to work beyond the nominal duty to compensate for the handicaps caused by the fault. Thus, it is important to reduce the overall system performance to an acceptable degraded performance. Moreover, in some faulty situation, the system cannot able to follow the command input and it is necessary to change it in order to continue the system operation. changing the performance can be achieved by changing the objective function of MPC [11],[12] or it can also determined by selecting a predesigned reference model [6]. In our proposal, we combine both methods. Where the objective function is changed according to DSM value in order to find a feasible solution of a constrained MPC and a degraded reference model can be used to reduce the effort to find a feasible solution for constrained MPC. Multi-reference and command control Block is important to select an acceptable degraded performance in case of a specified fault in addition to, the selection of a new command input in case of some faulty situation in order to maintain the process operation availability. The design of degraded reference model and command input for actuator faults is addressed in [6]. This method can be generalized in most of the faulty case. This Block is activated according to the information received from supervisory controller.

B. MPC Employing DSM Block

In the previous section, we discussed the MPC with constraints and showed how DSM can be handled in MPC. In most faulty system, the information from FDI is not accurate and sufficient to complete the fault description. That is because of the nonlinearities, uncertainties and/or disturbance in the system model. Hence, operating MPC with DSM is important to insure safety operation of the system and to compensate the missed information about the fault and uncertainties in the faulty model. DSM index is used to specify the priority of the constraints to find a feasible solution and to change the objective function parameters (weights).

C. Multi-model FDI and State and/or Parameter Estimation

Most of the model-based fault diagnosis systems in the literature compare either the actual signals or the identified system parameters, according to the fault type, with the estimated from faulty models to generate the residual ([20]). Designing a residual generation system, which is insensitive to model parameter variations and external disturbances, is a formidable task and it is called a robust FDI system. In [2], it is explained, how DSM can be helpful in designing a robust FDI system. This method for fault isolation depends on the generation of DSM from a set of models defined as $M = \{M_0, M_1, \ldots, M_z\}$ of the system under consideration (analytical redundancy of DSM) and compares the generated DSM with the actual value calculated from the measured state. Each one of these models simulates one fault of the faults set, which should be isolated in addition to the nominal fault free model. Multi-model FDI systems were addressed in different work (see for example [20]-[22]). The main difference between [2] and the other works is the use of DSM in the comparison between the faulty model and the actual model instead of the state directly. The model is described in general as

$$M_i : \begin{cases} \mathbf{x}_i(k+1) = g_i(\hat{\theta}_i(k), \hat{\mathbf{x}}_i(k), \mathbf{u}(k), \mathbf{f}_i(k)) \\ \mathbf{y}_i = \mathbf{C}\,\hat{\mathbf{x}}_i(k) \end{cases} \tag{26}$$

with $i = 0,1,..., z$.

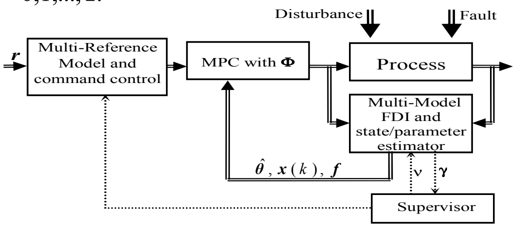

Disturbance
Fault
r
Multi-Reference Model and command control
MPC with Φ
Process
Multi-Model FDI and state/parameter estimator
$\hat{\theta}$, $x(k)$, f
ν
γ
Supervisor

Fig. 2. Overall structure of proposed FTC system

where $\mathbf{x}_i \in \Re^n$ is the state vector of the system model i, $\boldsymbol{u} \in \Re^{\mathrm{m}}$ the input vector; $\boldsymbol{y}_i \in \Re^p$ output vector; $\boldsymbol{f} \in \Re^l$ unknown additive fault signal vector; g_i: $\Re^n \times \Re^m \times \Re^l \to \Re^n$, $\hat{\boldsymbol{\theta}}_i$ system parameters for faulty model i; $\boldsymbol{C} \in \Re^{n \times p}$ constant matrix; and z number of anticipated fault. $i = 0$ means fault free case, nominal model.

The fault diagnosis and isolation subsystem is activated when $\delta(t) < 0$ and/or $d\delta(t)/dt < 0$. $d\delta(t)/dt < 0$ means that the state trajectory moves in the direction of unsafe operation. In general, faults can be divided into two types: additive faults, which can be simulated as an unknown external signal, and multiplicative faults, which represent the change in the parameters of the system. In both cases, it is necessary to estimate the unknown external input or new system parameters, according to the fault type, in order to obtain information about the fault. Therefore, parameters and state estimation are considered as a subsection of the FDI system. The output of that Block is the estimated state and faulty model parameters, which are submitted to MPC Block. It also sends status signal to the supervisory Block. As accurate the estimated data and fault diagnosis, a good performance recovery can be obtained.

D. Supervisory Block

Based on the results of FDI Block, the fault information is positive or negative. Positive information means that one of the faulty models can describe the fault. Negative information signifies that it is difficult to represent the fault by one model of the set. Thus, positive information is treated according to the scenario of fault recovery described before and the reference model and command signal can be selected easily according to the history of the system operation and the operation experience. Contrarily, negative information is not easy to be handled and therefore the supervisory controller should select the command input as well as the reference model and/or reconfigure the system in order to maintain the system availability. The supervisor receives the data γ from FDI Block, which contains fault type, output performance index and DSM value and it sends the signal ν to FDI in order to select the new model, which has to be used in the MPC. DSM plays an important role in supervisory control. It can be considered as a safety index for the recovery control performance, which is described by MPC. It is used in adapting the command input signal and in configuring the reference models.

V. Experimental Results

The above algorithms are tested in real-time operation on an experimental laboratory process described in [3]. The process, shown in Fig. 3, consists of two-tanks. Each tank has a control valve at the output line in order to control its level. In the current experiment the interconnecting valve was fully opened, the load valve (control valve of 2nd tank) was fixed at 10% to simulate the load or disturbance; the valve, of the left tank, was used to control the level. The two-tank system was fed at constant flow of 1 l/s in the first tank. The discrete linear model for the two-tank system at sampling rate equal to 10 Hz without load/disturbance (0% load valve) is given in Table 1.

Table 1: Linear state-space model of the two-tank-system

A	**B**
$\begin{bmatrix} 0.9748 & 0.0019 & -0.0146 \\ -0.1616 & -0.2104 & 0.5555 \\ -2.4323 & -1.1408 & 0.2307 \end{bmatrix}$	$\begin{bmatrix} -0.0004 \\ -0.0105 \\ -0.0173 \end{bmatrix}$
C	**D**
$[1 \quad 0 \quad 0]$	$[0]$

The output h is the level in the tank [m] and the input is the valve opening.

Consider that the variables relevant to system safety are the tank level rate (dh/dt) and v_i [m], the valve limb movement, which simulates the valve opening. The safe operation region **Φ** is given by:

$$dh/dt + 0.8\, v_i - 0.08 < 0;\ dh/dt + 0.75\, v_i + 0.14 > 0;$$
$$-0.4 < dh/dt < 0.4;\ -0.5 < v_i < 0.5, \tag{27}$$

where the valve opening is normalized within [-0.5, 0.5] i.e. 0.5 means fully opened and -0.5 completely closed. The level rate changes (dh/dt) in [mm/s].

A model predictive controller, corresponding to Section IV, is used with and without DSM to regulate the level of the left tank at the set point of 0.3 m in case of actuator fault. Fig. 4 shows real-time implementation of the above algorithm for the two-tank system in case of bias fault 30% in the control valve after 500 s until 1500 s (fault scenario). MPC without DSM is used as nominal controller from the starting until fault occurred. As shown in Fig. 5, the DSM (Fig. 5b) is negative after fault; according to FTC algorithm discussed another MPC with DSM constraints is used until the fault is identified. In this experiment, it is assumed that no information available about the fault (Remark 1); therefore, the second controller (MPC with constraints) has been used alone to recover the performance. It is clear that the output performance (Fig. 5a) has improved using the second controller and the DSM value as well.

Fig. 6 shows the real-time results in case of repeated 20% actuator bias fault two-time between "350:500s" and "700:920sec". The nominal controller has been used from the start time (t=0) until the second fault (t≥700) i.e. the nominal controller has been used to recover performance in the first fault. MPC with constraints is used to recover the second fault after DSM<0 as in Fig. 6. It is clear that DSM with constraints has improved the system performance and the safety margin better than nominal MPC. Fig. 4 shows real-time implementation of the above FTC algorithm for the two-tank system in case of bias fault 20% in the control valve after 320s until 480s (fault scenario). Three controller are used MPC without DSM in normal operation until DSM<0 (0:370sec), MPC with DSM constraints when DSM<0 until fault diagnosis (370:400s), and MPC with DSM using faulty model (actuator fault model) after fault diagnosis (after 400s). It is clear that the output and safety performance are better than Fig. 5 and Fig. 6 using two controllers only; the recovery time is shorter, the steady state error is smaller and DSM is better than the previous results for the same fault. It is clear that, more accurate FDI system improve the overall system performance.

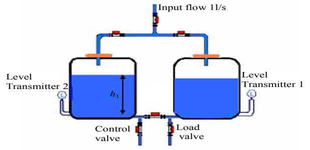


Fig. 3. Schematic diagram of a two-tank system

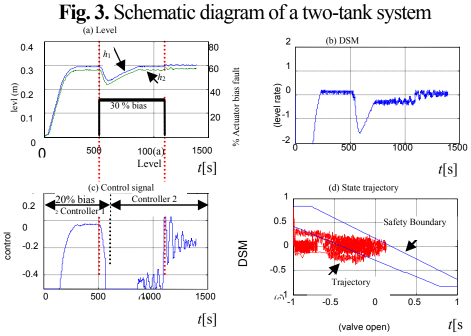
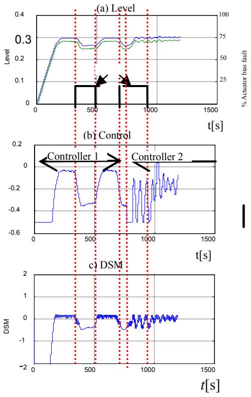


Fig. 4. FTC response using two predictive controller without and with DSM



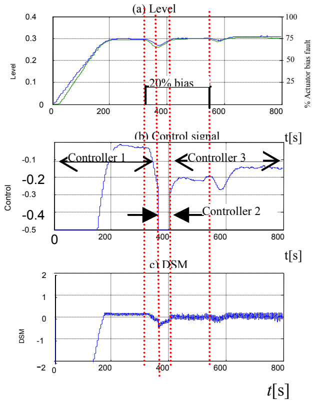


Fig. 5 FTC response using one and two controllers

Fig. 6 FTC response using three controllers

VI. Conclusions

MPC using DSM is discussed in the application of FTC system. The controller design based on DSM improves safety-assessment of safety-critical systems. DSM index is also used in changing the objective function of MPC in order to find a feasible solution of MPC and satisfy the safety requirements for a predefined performance. Results of a real-time implementation on a two-tank process demonstrate the advantage of the proposed approach for FTC. Since a controller design based on DSM can compensate for the nonlinearities in the system model, the proposed FTC method could be applied in nonlinear system. Since the practical applications of MPC with constraints are limited due to the high computation burden, the considering of DSM in adapting MPC parameters simplify the algorithm of MPC with constraints. Hence, the proposed algorithm could be applied in large scale system. There are some open topics in applying this approach, which should be covered in the future: For example, DSM measuring for large-scale system and using DSM in building an acceptable degraded reference. Hence, all these topics will be undertaken in the future work as well as the problem of applying DSM to fault prognosis.

References

[1] E. Badreddin and M. Abdel-Geliel. "Dynamic safety margin principle and application in control of safety critical system" *IEEE International Conference of Control Application* (CCA 2004) *Conference*, September 2-4, 2004, Taiwan. pp 689-695

[2]. M. Abdel-Geliel and E. Badreddin, "Dynamic safety margin in fault diagnosis and isolation". *European Safety and Reliability (ESREL) conference*, June 27-30, 2005, Tri city Poland, pp. 1-6

[3] M. Abdel-Geliel and E. Badreddin, "Adaptive controller using dynamic safety margin for hybrid laboratory plant". *American Control Conference 2005*, Portland, Oregon, USA., pp. 1443-1448

[4] M. Blanke, M. Staroswieki and N. Eva Wu, "Concept and methods in fault-tolerant control", *Tutorial at American Control Conference*, June 2000.

[5] M. Mahmoud, J. Jiang, and Y. Zhang, "*Active Fault Tolerant Control System.*" *Lecture Notes in Control and information Sciences*", Springer, 2003.

[6] Y. Zhang and J. Jiang, "Fault tolerant control system design with explicit consideration of performance degradation". *IEEE Trans. in Aerospace and electronic system*, vol. 39, June 2003.

[7] E.F. Camacho and C. Bordons, "*Model Predictive Control*". Springer, 1999.

[8] J.A Rossiter, "*Model-Based Predictive Control: Practical Approach*", CRC, 2003.

[9] J.M. Maciejowski, "*Predictive Control with Constrained*", Prentice Hall, 2001.

[10] D. Q. Mayne, J. B. Rawlings, C. V. Rao and Po. M. Scokaert, "Constrained model predictive control: Stability and optimality", *Automatica*, 36, pp 789-814, 2000.

[11] J.M. Maciejowski, "Reconfiguring Control system by optimization in European Control Conference ECC, Brussel, 1997

[12] J. M. Maciejowskiand C.N. jones, "MPC fault-tolerant flight control case study: Flight 1862". *IFAC Safe Process Conference*, Washington DC, pp 9-11, June 2003

[13] F. Borrelli, *"Constrained Optimal Control of Linear and Hybrid Systems"*, Lecture Notes in Information Science, Springer, 2003.

[14] J. Vada, O. Slupphaug T.A. Johansen and B.A. Foss, "Linear MPC with optimal prioritized infeasibility handling: Application computation issues and stability", *Automatica*, 37, pp 1835-1843, 2001.

[15] E. Kerrign and J. Maciejowski, "Designing model predictive controller with prioritised constraints and objectives", *IEEE Conf. on Computer Aided control System and Design*, Sept 2002.

[16] A. Bemporad, M Morari, V. Dua and E.N. Pistikopoulos, "The explicit linear quadratic regulator for constrained system", *Automatica*, 38, pp 3-20, 2002.

[17]A. Gambier and H. Unbehauen, "Multivariable generalized state-space receding horizon control in a real-time environment". *Automatica*, 35, pp. 1787-1797, 1999.

[18]M. Diehl, J. Björnberg, "Robust dynamic programming for min-max model predictive control of constrained uncertain system". *IEEE Trans. on Automatic Control*, 49, December 2004.

[19] Eric Kerrigen and I. Maciejowski, "Feedback min-max MPC using a single linear program: Robust stability and explicit solution". *International Journal of Robust and Nonlinear Control*, 14, 395-413, 2004

[20]J. Patton, P. Frank, R. Clark 2000. *Issues of Fault Diagnosis for Dynamic System*, Springer, 2000

[21] Y. Zhang and X Z. Rong Li, "Detection and Diagnosis of sensor and Actuator failures Using IMM Estimator", *IEEE Tran. In Aerospace and electronic system*, vol 34, no 4, 1998

[22]A. Alessandri, T. Hawkinso, A.J. Healey and G. Veruggio, "Robust model-based fault diagnosis for unmanned underwater vehicles using sliding mode-observers". Proc. of 11th *Int. Symposium on Unmanned Untethered Submersible Technology* (UUST'99), August 22-25, 1999.

[23] M. Abdel-Geliel, E. Badreddin and A. Gambier, "Dynamic safety margin in fault tolerant model predictive controller". *IEEE International Conference of Control Application* (CCA 2005), *pp. 803-808*, September, 2005, Canada.

[24] F. Blanchini, "Set invariance in control", Automatica, 35 (1999), pp. 1747-1767.