CHAPTER 5

ADAPTIVE DYNAMIC MATRIX CONTROLLER

5.1 INTRODUCTION

MPC has established itself in industry as an important form of advanced multivariable control (Townsend and Irwin, 2001). Since, the advent of MPC, various model predictive controllers have evolved to address an array of control issues (García et al 1989; Froisy 1994). DMC (Cutler and Ramaker 1980) is the most popular MPC algorithm used in the process industry today (Qin and Badgwell 1996; Townsend et al 1998). A major part of DMC’s appeal in industry stems from the use of a linear finite step response model of the process and a quadratic performance objective function. The objective function is minimized over a prediction horizon to compute the optimal controller output moves as a least squares problem.

Control techniques such as DMC, Model Algorithmic Control (MAC), Internal Model Control (IMC) and Inferential Control explicitly use a process model. Of these, undoubtedly the most popular with the process industry is DMC. The algorithm has been successfully applied in many cases were the conventional PID type control is unsuitable for various reasons. Several application examples of DMC have been reported (Qin and Badgwell 1996; Townsend et al 1998). The modeling philosophy and the ability of DMC in handling complex control problems commonly encountered in multivariable systems (nonlinear MIMO system with complex coupling dynamics and input-output constraints) have made it very popular algorithm.
It performs well when applied to complicated nonlinear systems, and even systems for which an analytical model is not available. When compared with other models, obtaining a step response model for any complex industrial processes will be easier. Also, the step response model can be obtained online.

A model predictive control algorithm optimizes closed loop performance for a nominal operating point. However, as the process moves away from this point, control usually becomes sub-optimal due to process non-linearity. An adaptive DMC could be expected to perform well in the presence of uncertainties, non-linearity, time-varying process parameters and complex coupling dynamics.

A more practical adaptive strategy uses a gain and time constant schedule for updating the process model (McDonald and McAvoy 1987; Chow et al 1998). An extension of this method is to use multiple models to update the process model. Linear models that described the system at various operating points are developed based on plant measurements. Past researchers (Banerjee et al 1997) have illustrated that linear models can be combined in order to obtain an approximation of the process that approaches its true behavior. Two different multiple model controller design methods can be employed to maintain the performance of the controller overall operating levels. In one case, a controller is designed for each level of operation. Using gain scheduling adaptive control scheme, one of the controller is selected based on operating point. In the past, this methodology has been applied to generalized predictive control and proportional-integral-derivative controllers. In the second case, the controller moves are then weighted based on the prediction error calculated for each controller. The resulting weights are obtained using recursive identification such that the prediction error is minimized (Yu et al 1992; Schott and Wayne Bequette 1994).
Townsend et al (1998) developed a nonlinear DMC controller that replaces the linear process model with a local model network. This local model network contains local linear ARX models and is trained using a hybrid learning technique. From this local model network, the DMC controller is supplied with a weighted step response model. Narendra and Xiang (2000) design multiple controllers using both fixed and adaptive process models. Based on the prediction error for each of these process models, a procedure is designed that switches between the controllers corresponding to the process model with the lowest prediction error. This allows the controller to incorporate both time-invariant dynamics along with time-varying dynamics.

Danielle Dougherty and Doug Cooper (2002) developed multiple model adaptive control for multivariable system in which final output is estimated as an interpolation of the individual controller outputs weighted based on the current value of the measured process variable. Zhong Zhaoa et al (2002) developed a nonlinear DMC based on multiple operating models. They have used switching algorithm to select suitable model and proved closed loop stability. Brian Auflerheide and Wayne Bequette (2003) suggested a method to extend DMC to handle different operating regimes and to reject parameter disturbances. They have implemented one single constrained optimizer which uses a weighted model bank as a prediction model. Gupta (1998), Isabel Guiamab and Michael Mulholland (2004) have suggested a novel DMC algorithm to control a TITO system with an integrating behaviour.

Theoretical studies are needed to address the issue of closed loop stability over the entire range of nonlinear operation. It has been reported by past researchers Greene and Willsky (1980) that the overall Multi Model Adaptive Control (MMAC) system may not be stable even if each individual controller is stable over the entire range of operation. In addition,
Narendra and Xiang (2000) address the issue of stability for linear time-invariant discrete systems using MMAC. The results included a proof of global stability for the overall system.

In this chapter, two different methods of MMAC strategy based multivariable DMC are developed for TCTILS. The first MMAC strategy based multivariable DMC uses gain scheduler for updating the controller based on the current value of the measured process variables. The second MMAC strategy uses Takagi-Sugino Fuzzy Inference System (T-S FIS) as a scheduler to estimate global controller output from each linear controller output. The proposed scheme can easily be expanded to more controllers if desired by the practitioner.

5.2 MULTIVARIABLE DYNAMIC MATRIX CONTROL

Multivariable DMC has been discussed extensively by past researchers (Cutler and Ramaker 1980; Marchetti et al 1983) and is summarized here.

The basic operation of DMC is shown in the Figure 5.1. The Dynamic Matrix Controller basically consists of two components

- Model Based Optimizer
- Model Based Predictor

The function of the Model Based Optimizer is to compute the control action for the length of the control horizon based on the difference between the reference trajectory and predicted trajectory of prediction horizon length. Though, the Model Based Optimizer calculates the control moves for the length of control horizon only the first move is taken into account and is applied to the process.
The function of the Model Based Predictor is to predict the process variable profile for the length of the prediction horizon using the effect of change in the control action on the process variable. This output prediction is again used by the Model Based Optimizer to produce the next control action.

5.2.1 Formulation of DMC Algorithm

The Dynamic Matrix Control is based on the step response model which is written in the form (Wayne Bequette 2008)

\[
\hat{y}_k = \sum_{i=1}^{N-1} S_i \Delta u_{k-i} + S_N u_{k-N}
\]

(5.1)

where \( \hat{y}_k \) is the model prediction at time step \( k \), \( S_i \) is step response coefficient and \( u_{k-N} \) is the manipulated input \( N \) steps in the past. \( N \) is called as the Model Horizon which is the instant at which the step response reaches the steady state. The difference between the measured output \( (y_k) \) and the model prediction \( \hat{y}_k \) is called the additive disturbance \( d_k \)

\[
d_k = y_k - \hat{y}_k
\]

(5.2)
The “corrected prediction” \( \hat{y}_k^c \) is the equal to the actual measured output at step \( k \),
\[
\hat{y}_k^c = \hat{y}_k + \hat{d}_k
\]  \( (5.3) \)

Similarly the corrected predicted output for the \( j^{th} \) step into the future can be found as

\[
\hat{y}_{k+j}^c = \hat{y}_{k+j} + \hat{d}_{k+j}
\]

\[
\hat{y}_{k+j}^c = \sum_{i=1}^{j} S_i \Delta u_{k+i-j} + \sum_{i=j+1}^{N-1} S_i \Delta u_{k+i-j} + S_N \Delta u_{k-N+j} - \hat{d}_{k+j}
\]  \( (5.4) \)

Effect of future control moves \hspace{1cm} \text{effect of past control moves} \hspace{1cm} \text{correction term}

We can separate the effects of the past and future control moves as follows

\[
\hat{y}_{k+j}^c = S_1 \Delta u_{k+j} + S_2 \Delta u_{k+j-2} + \ldots + S_j \Delta u_k \quad \text{effect of current and future moves}
\]

\[
+ S_N \Delta u_{k-N+j} + S_{j+1} \Delta u_{k-1} + S_{j+2} \Delta u_{k-2} \quad \text{effect of past moves}
\]

\[
+ \ldots + S_{N-j} \Delta u_{k-N+j} \hspace{1cm} \text{correction term}
\]  \( (5.5) \)

In matrix-vector form, a prediction horizon of \( P \) steps and a control horizon of \( M \) steps yields (Prediction and Control Horizons are of the same significance as in MPC)
This can be written in matrix-vector notation as follows

\[
\hat{Y}_k^c = S_f \Delta u_f + S_{\text{past}} \Delta u_{\text{past}} + S_N u_p + \hat{d}
\]  

(5.7)
In Equation (5.7) the corrected predicted output response is naturally composed of the free response (Contribution of the current and the future control moves) and a forced response (The output changes that are predicted if there are no future control moves)

The difference between the set-point trajectory \( r \) and the future predictions is

\[
\begin{align*}
\hat{Y}^c - r = & \underbrace{r - \left[ S_{\text{past}} \Delta u_{\text{past}} + S_{\text{N}} u_{\text{p}} + \hat{d} \right]}_{\text{Corrected Error (if no current and future control moves were made), } E} - \underbrace{S_f \Delta u_f}_{\text{Predicted Error}} \\
\end{align*}
\] (5.8)

Equation (5.8) can be written as

\[
E^c = E - S_f \Delta u_f \\
\] (5.9)

The least-squares objective function is

\[
\phi = \sum_{i=1}^{p} (e^c_{k+i})^2 + w \sum_{i=0}^{M-1} (\Delta u_{k+i})^2 \\
\] (5.10)

The quadratic terms can be written in the matrix-vector form as

\[
\begin{align*}
\sum_{i=1}^{p} (e^c_{k+i})^2 &= \begin{bmatrix} e^c_{k+1} & e^c_{k+2} & \cdots & e^c_{k+p} \end{bmatrix} \begin{bmatrix} e^c_{k+1} \\
e^c_{k+2} \\
\vdots \\
e^c_{k+p} \end{bmatrix} = (E^c)^T E^c \\
\end{align*}
\] (5.11)

and
The control move formulated above is for the case of single variable process (SISO). Extending the same for the multi variable process (example TITO process) requires four step response coefficients to consider the effect of each input on each process variable and two different sets of weights which are the move suppression weights ($\gamma$) and the controlled
variable weights ($\lambda$). These weights are used for the multi variable process in the matrix form (diagonal matrix).

The output from the step response coefficients is used as given by Equations (5.16) and (5.17).

$$y_1(k) = \sum_{i=1}^{N} a_i \Delta u_1(k-i) + a_{N+1} u_1(k-N-1) + \sum_{i=1}^{N} b_i \Delta u_2(k-i) + b_{N+1} u_2(k-N-1) + d_1(k)$$

(5.16)

$$y_2(k) = \sum_{i=1}^{N} c_i \Delta u_1(k-i) + c_{N+1} u_1(k-N-1) + \sum_{i=1}^{N} d_i \Delta u_2(k-i) + d_{N+1} u_2(k-N-1) + d_2(k)$$

(5.17)

where, $a_1 \ldots a_N$ are the step response coefficients of the process variable 1 obtained by applying a step input to the input variable 1.

$b_1 \ldots b_N$ are the step response coefficients of the process variable 1 obtained by applying a step input to the input variable 2.

$c_1 \ldots c_N$ are the step response coefficients of the process variable 2 obtained by applying a step input to the input variable 1.

$d_1 \ldots d_N$ are the step response coefficients of the process variable 2 obtained by applying a step input to the input variable 2.

The corresponding disturbance models is as given by Equations (5.18) and (5.19):

$$d_1(k) = y_1^{\text{meas}}(k) - y_1^{\text{mod el}}(k)$$

(5.18)

$$d_2(k) = y_2^{\text{meas}}(k) - y_2^{\text{mod el}}(k)$$

(5.19)
The Dynamic matrix for the multi variable (TITO) process is obtained by the coefficients of the four step response models from the plant. The Dynamic matrix for a TITO is of the dimensions 2P x 2M and is represented as follows

\[
S_f = \begin{bmatrix}
a_1 & 0 & \cdots & 0 & b_1 & 0 & \cdots & 0 \\
a_2 & a_1 & \cdots & 0 & b_2 & b_1 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
a_M & a_{M-1} & \cdots & a_1 & b_M & b_{M-1} & \cdots & b_1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
a_p & a_{p-1} & \cdots & a_{p-M+1} & b_p & b_{p-1} & \cdots & b_{p-M+1} \\
c_1 & 0 & \cdots & 0 & d_1 & 0 & \cdots & 0 \\
c_2 & c_1 & \cdots & 0 & d_2 & d_1 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
c_M & c_{M-1} & \cdots & c_1 & d_M & d_{M-1} & \cdots & d_1 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
c_p & c_{p-1} & \cdots & c_{p-M+1} & d_p & d_{p-1} & \cdots & d_{p-M+1}
\end{bmatrix}
\]

DYNAMIC MATRIX

The diagonal matrix formed by using the move suppression weight ($\gamma$) is represented by the notation ($\Gamma$) which equals

\[
\Gamma = \text{diag}(\gamma_1, \ldots, \gamma_2, \ldots, \ldots, \gamma_n, \ldots)
\]

Length P

where n represents the number of the manipulated variable. This is a diagonal matrix formed by the input suppression weight for each input for the length of the prediction horizon. Thereby the dimension of the move suppression matrix is nP x nP. The diagonal matrix formed by using the control variable weights ($\lambda$) is represented by the notation ($\Lambda$) which equals
\[ \Lambda = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n) \]

This is a diagonal matrix formed by the control variable weight for each input for the length of the control horizon. Thereby the dimension of the control variable weight matrix is \( nM \times nM \). Thus the system of Linear equations is written in a compact form as shown:

\[ y^{\text{lin}} = S_f \Delta u + y^{\text{past}} + d \]

(5.20)

The optimization problem for the TITO case is:

\[ \begin{align*}
\min_{\Delta u} & \quad (y^{\text{sp}} - y^{\text{lin}})^T \Gamma (y^{\text{sp}} - y^{\text{lin}}) + \Delta u^T \Lambda^T \Lambda \Delta u \\
\end{align*} \]

(5.21)

The solution for the optimization problem is

\[ \Delta u = (S_f^T \Gamma^T \Gamma S_f + \Lambda^T \Lambda)^{-1} S_f^T \Gamma^T \Gamma (y^{\text{sp}} - y^{\text{past}} - d) \]

\[ K \]  
error vector \( \delta \)

(5.22)

The control move formulated above is for the case of MIMO process.

### 5.3 FORMULATION OF MMAC STRATEGY FOR TCTILS USING MULTIVARIABLE DMC

When DMC is employed on nonlinear chemical processes, the application of this linear model based controller is limited to relatively small operating regimes. Specifically, if the computations are based entirely on the model prediction (i.e., no constraints are active), the accuracy of the model has significant effect on the performance of the closed loop system.
(Gopinath et al 1995). Hence, the capabilities of DMC will degrade as the operating level moves away from the original design level of operation.

To maintain the performance of the controller over a wide range of operating levels, a MMAC strategy for DMC has been developed. While MMAC will not capture severe nonlinear dynamic behavior, it will provide significant benefits over linear controllers. The work focuses on a MMAC strategy for processes that are stationary in time, but nonlinear with respect to the operating level. This method is not applicable to processes where the gain of the process changes sign. The method of approach is to construct a small set of DMC process models that span the range of expected operation. By combining the process models to form a nonlinear approximation of the plant, the true plant behavior can be reasonably achieved (Banerjee et al 1997). If linear process models and controllers are employed, the wealth of design and tuning strategies for the linear controllers can be used. This is a benefit to the control practitioner since they do not have complete knowledge of the nonlinear control strategies currently available in the literature (Schott and Wayne Bequette 1994; Townsend et al 1998). The accuracy of the nonlinear approximation can be increased by combining more models. However, this is expensive because each model requires the collection of plant data at a different level of operation. The number of DMC process models ultimately employed is a practical determination made by the control practitioner on a case-by-case basis. In most cases, the practitioner will balance the expense of collecting data with the need to improve the nonlinear approximation. Each controller has their own step response model that describes the process dynamics at a specific level of operation.
The novelty of this work lies in designing and combining multiple linear DMC controllers using Takagi-Sugeno Fuzzy Inference System (TS-FIS). Based on the values of the measured process variables, TS-FIS generates weights and global output of the controller is obtained by calculating average weighted sum of all the linear controller outputs. The parameters for each controller are tuned using real coded GA. The result is a simple and easy to use method for adapting the control performance without increasing the computational complexity of the control algorithm.

5.4 DESIGN OF GAIN SCHEDULING ADAPTIVE DMC

For comparison, a gain scheduling adaptive DMC (GSA-DMC) controller is designed and present alongside the Nonlinear DMC (NDMC) method. The following steps describe the design procedure for determining the tuning parameters for GSA-DMC scheme. The step response coefficients and tuning parameters are calculated offline prior to the startup of the GSA-DMC controller and remain constant during operation.

**Step 1:** computes linearised regimes and Design Level of Operation (DLO) for TCTILS. Based on the insight of TCTILS, its entire non linear operating regime is divided into three linearised operating regimes for each process variable as shown in Figures 5.2 and 5.3 respectively. Fuzzy C-Mean clustering technique is used to obtain linearised regimes for TCTILS. The operating range for each linearised regime of $h_1$ and $h_2$ are presented in the Table 5.1. The midpoint of each operating regime is considered as DLO and is presented in Table 5.2. Three DMC controllers are designed at each pair of DLO. Only three linear DMC controllers are used in this work because as mentioned previously, collecting step response plant data is difficult and time consuming. The method can easily be expanded to more linear DMC controllers if desired by the practitioner.
Figure 5.2  Linearised regimes of process variable $h_1$

Figure 5.3  Linearised regimes of process variable $h_2$
Table 5.1 Operating range for each linearised regime in TCTILS

<table>
<thead>
<tr>
<th>Linearised Regimes</th>
<th>Process variable1 $h_1$</th>
<th>Process variable2 $h_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$F_{IN1}$ (LPH)</td>
<td>$h_1$ (cm)</td>
</tr>
<tr>
<td>Lower Region</td>
<td>0 - 164.5</td>
<td>0 - 9.062</td>
</tr>
<tr>
<td>Middle Region</td>
<td>164.5 - 337.5</td>
<td>9.062 - 25.66</td>
</tr>
<tr>
<td>Upper Region</td>
<td>337.5 - 500</td>
<td>25.66 - 50</td>
</tr>
</tbody>
</table>

Table 5.2 DLO for each Dynamic Matrix Controller in TCTILS

<table>
<thead>
<tr>
<th>Linearised Regimes</th>
<th>Dynamic Matrix Controllers</th>
<th>Process variable1 $h_1$</th>
<th>Process variable2 $h_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$F_{IN1}$ (LPH)</td>
<td>$h_1$ (cm)</td>
<td>$F_{IN2}$ (LPH)</td>
</tr>
<tr>
<td>Lower Region</td>
<td>DMC1</td>
<td>82.25</td>
<td>4.531</td>
</tr>
<tr>
<td>Middle Region</td>
<td>DMC2</td>
<td>251</td>
<td>17.361</td>
</tr>
<tr>
<td>Upper Region</td>
<td>DMC2</td>
<td>418.75</td>
<td>37.83</td>
</tr>
</tbody>
</table>

Step 2: involves the selection of a sample time, $T$: The value of $T$ is given by Equation (5.23) which balances the desire for a low computation load (a large $T$) with the need to properly track the evolving dynamic behavior (a small $T$). Too slow of a sampling rate will lead to information losses, and too fast of a sampling rate could lead to numerically sensitive procedures as confirmed by Ljung (1987). Since many control computers restrict the choice of $T$ (Franklin and Powell 1980; Åström and Wittenmark 1984), the value of $T$ can be selected as other than the recommended value given by Equation (5.23). The time constant and delay time of each sub-process in various regimes of TCTILS are presented in Table 5.3.
Table 5.3 Time constant and delay time of the sub-process in various regimes of TCTILS

<table>
<thead>
<tr>
<th>Linearised Regimes</th>
<th>Time constant $\tau_{rpvsu}$ (seconds)</th>
<th>Delay time $\theta_{rpvsu}$ (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$G_{11}$</td>
<td>$G_{12}$</td>
</tr>
<tr>
<td>Lower Region</td>
<td>1.7</td>
<td>1.80</td>
</tr>
<tr>
<td>Middle Region</td>
<td>8.1</td>
<td>10.5</td>
</tr>
<tr>
<td>Upper Region</td>
<td>49.8</td>
<td>65</td>
</tr>
</tbody>
</table>

$T_{rpvsu} = \max(0.1\tau_{rpvs}, 0.5\theta_{rpvsu})$  
$r_{pv} = 1,2,\ldots,R_{pv}; \ s_{u} = 1,2,\ldots,S_{u}$

$T = \min(T_{rpvsu})$  \hspace{1cm} (5.23)

where

- $T$ – Sample time
- $\tau_{rpvsu}$ – Time constant of sub-process relating the $s_{u}$\th controller output to the $r_{pv}$\th process variable at the DLO
- $\theta_{rpvsu}$ – Dead time of sub-process relating the $s_{u}$\th controller output to the $r_{pv}$\th process variable at the DLO
- $r_{pv}$ – $r$\th Process variable (for TCTILS $r_{pv}=1,2$)
- $s_{u}$ – $s_{u}$\th Controller output (for TCTILS $s_{u}=1,2$)

Sampling time $T$ for lower region = $\min(T_{rpvsu})$

$= \min(\max[0.17,0.2], \max[0.18,0.3], \max[0.17,0.2], \max[0.18,0.3])$

$= \min(0.2,0.3,0.2,0.3) = 0.2$ second

Sampling time $T$ for middle region = $\min(T_{rpvsu})$

$= \min(\max[0.81,0.5], \max[1.05,0.6], \max[0.81,0.5], \max[1.05,0.6])$

$= \min(0.81,1.05,0.81,1.05) = 0.81$ second
Sampling time $T$ for upper region = $\min(T_{\text{pvsu}})$

$= \min(\max[4.98,0.6], \max[6.5,0.65], \max[4.98,0.6], \max[6.5,0.65])$

$= \min(4.98,6.5,4.98,6.5) = 4.98$ second

In the present work, the sampling time for all the three DMC controllers is chosen as $T = 0.1$ second.

**Step 3:** computes the prediction horizon, $P$, and the model horizon, $N$, in terms of samples as the settling time of the slowest sub-process in the multivariable system (Shridhar and Cooper (1997, 1998)). A larger $P$ improves the nominal stability of the closed loop control of the multivariable system. For this reason, $P$ is calculated such that it includes the steady-state effect of all past controller output moves, i.e. it is calculated as the open loop settling time of the slowest sub-process in the multivariable system. The value of $N$ also to be equal to the open loop settling time of the slowest sub-process to avoid truncation error in the predicted process variable profiles. This is required for $N$ to be long enough to avoid the instabilities that can otherwise result since truncation of the model horizon misrepresents the effect of controller output moves in the predicted process variable profile (Lundström et al 1995).

![Step response of slowest sub-process of the TCTILS showing prediction horizon (P) =settling time = 630-300=330](image)

**Figure 5.4** Step response of slowest sub-process of the TCTILS (showing prediction horizon (P) =settling time = 630-300=330)
The open loop step response of the slowest sub-process of TCTILS which is $G_{12}$ of the upper region is shown in Figure 5.4. From the Figure 5.4, the prediction horizon is estimated and it is equal to settling time of the step response which is given as

\begin{align*}
\text{Prediction horizon} & \quad P = 330 \\
\text{Model horizon} & \quad N = 330
\end{align*}

**Step 4:** computes the control horizon, $M$, equal to 63.2\% of the settling time of the slowest sub-process in the multivariable system. This is required for $M$ to be long enough such that the results of the control actions are clearly evidenced in the response of the multivariable system.

\begin{align*}
\text{Control horizon } M &= 63.2\% \text{ of the settling time of the slowest} \\
&\quad \text{sub-process in TCTILS} \\
&= 63.2\% \text{ of } 330 \\
&= 209
\end{align*}

**Step 5:** computes controlled variable weights, $\gamma$, and move suppression coefficients, $\lambda$. In most of the multivariable process, the controlled variable weights are set equal to one. However, the practitioner is free to select these values to recast the measured process variables into the same units. Or, the practitioner can use these parameters to achieve tighter control of a particular measured process variable by selectively increasing its relative weight. The primary role of move suppression coefficients, $\lambda$, in multivariable DMC is to suppress aggressive controller actions. When the control horizon $M$ is 1, no move suppression coefficient is needed ($\lambda = 0$). If the control horizon $M$ is greater than 1, then the move suppression coefficients are needed.
In the present work, these values are tuned optimally using real coded GA. The sequence of steps involved in tuning of parameters of GSA-DMC scheme using real-coded GA is shown in Figure 5.5.

![Flowchart for tuning GSA-DMC using real coded GA]

**Figure 5.5 Flowchart for tuning GSA-DMC using real coded GA**

The decision variables of GSA-DMC scheme and parameters of Real-coded GA used for tuning are listed in the Table 5.4.
Table 5.4 Decision variables of GSA-DMC and real coded GA parameters to obtain optimal GSA-DMC parameters for TCTILS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>No of decision variables</td>
<td>12</td>
</tr>
<tr>
<td>- Move suppression coefficients</td>
<td>$\lambda_1, \lambda_2$ and Control variable weights $\gamma_1, \gamma_2$ for DMC1, DMC2 and DMC3</td>
</tr>
<tr>
<td>No. of population</td>
<td>75</td>
</tr>
<tr>
<td>Termination criterion</td>
<td>20 epochs</td>
</tr>
<tr>
<td>Mutation method</td>
<td>Non-uniform mutation</td>
</tr>
<tr>
<td>Crossover method</td>
<td>Arithmetic crossover</td>
</tr>
<tr>
<td>Mutation rate</td>
<td>0.15</td>
</tr>
<tr>
<td>Crossover rate</td>
<td>0.85</td>
</tr>
<tr>
<td>Selection method</td>
<td>Roulette wheel method</td>
</tr>
<tr>
<td>Objective function</td>
<td>$J = \frac{1}{2} \left[ \int_0^1 e_1^2 dt + \int_0^1 e_2^2 dt \right]$</td>
</tr>
</tbody>
</table>

The evolution of performance index $J$ value during various generations in real coded GA based tuning process of GSA-DMC is shown in Figure 5.6. The performance index $J$ is converged in $9^{th}$ generation. In this tuning process, performance index $J$ which is an average ISE of $h_1$ and $h_2$ is reduced to 783.2. The evolution of some of the controller parameters during various generations in real coded GA based tuning process of GSA-DMC is shown in Figure 5.7.

Figure 5.6 Evolution of performance index $J$ during tuning of GSA-DMC using real coded GA
Figure 5.7 Evolution of various parameters during tuning of GSA-DMC using real coded GA  a) $\hat{\lambda}_1$ for DMC1, b) $\hat{\lambda}_2$ for DMC2, c) $\gamma_1$ for DMC1, d) $\gamma_2$ for DMC1, e) $\gamma_1$ for DMC2 and f) $\gamma_2$ for DMC2
Figure 5.7 (Continued)
The real coded GA tuned controller parameters for GSA-DMC scheme are presented in Table 5.5.

**Table 5.5  Real coded GA tuned controller parameters for GSA-DMC scheme**

<table>
<thead>
<tr>
<th>DMC parameters</th>
<th>DMC1</th>
<th>DMC2</th>
<th>DMC3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sampling Time</td>
<td>0.1 second</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prediction Horizon</td>
<td>330</td>
<td>330</td>
<td>330</td>
</tr>
<tr>
<td>Control Horizon</td>
<td>209</td>
<td>209</td>
<td>209</td>
</tr>
<tr>
<td>Model Horizon</td>
<td>330</td>
<td>330</td>
<td>330</td>
</tr>
<tr>
<td>Move Suppression coefficient</td>
<td>( \lambda_1 )</td>
<td>0.627</td>
<td>0.597</td>
</tr>
<tr>
<td></td>
<td>( \lambda_2 )</td>
<td>1.349</td>
<td>1.126</td>
</tr>
<tr>
<td>Control Variable weights</td>
<td>( \gamma_1 )</td>
<td>2266</td>
<td>64.63</td>
</tr>
<tr>
<td></td>
<td>( \gamma_2 )</td>
<td>2006</td>
<td>57.78</td>
</tr>
</tbody>
</table>

**Step 6:** collects the unit step response coefficients, \( a_{rpvuis} \) (\( i = 1,2,\ldots,N; \ r_{pv} = 1,2,\ldots,R_{pv}; \ s_u = 1,2,\ldots,S_u \)), for controller output \( s \) on measured process variable \( r \). For this, the process data is generated by introducing a positive step in one controller output with the process at steady state and all the controllers in manual mode. In addition, all other controller output variables must remain constant. From the instant the step change is made, the response of each process variable is recorded as it evolves and settles at a new steady state. For a step in the controller output of arbitrary size, the response data is normalized by dividing through by the size of the controller output step to yield the unit step response. This is performed for each controller output to measured process variable pair, and it is necessary to make the controller output step large enough such that noise in the process variable measurement does not mask the true process behaviour.
For a TCTILS (TITO system), the multivariable dynamic matrix, \( A \), is formulated using the first \( P \) step response coefficients:

\[
A = \begin{bmatrix}
A_{11} & A_{12} \\
A_{21} & A_{22}
\end{bmatrix}_{2P \times 2M}
\]  

(5.24)

where \( A_{11} \) is constructed from the step response of process variable 1 (\( h_1 \)) obtained by a step change in controller output 1 (\( F_{IN1} \)). \( A_{ij} \) is given by

\[
A_{ij} = \begin{bmatrix}
a_{ij,1} & 0 & 0 & \cdots & 0 \\
a_{ij,2} & a_{ij,1} & 0 & \cdots & 0 \\
a_{ij,3} & a_{ij,2} & a_{ij,1} & \ddots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
a_{ij,M} & a_{ij,M-1} & a_{ij,M-2} & \cdots & a_{ij,1} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
a_{ij,P} & a_{ij,P-1} & a_{ij,P-2} & \cdots & a_{ij,P-M+1}
\end{bmatrix}_{P \times M}
\]  

(5.25)

A vector of (SM) controller output moves is computed over the control horizon and is given in Equation (5.26).

\[
\Delta \bar{u} = \begin{bmatrix}
\Delta u_1(n) \\
\Delta u_1(n+1) \\
\Delta u_1(n+2) \\
\vdots \\
\Delta u_1(n+M-1) \\
\Delta u_2(n) \\
\Delta u_2(n+1) \\
\Delta u_2(n+2) \\
\vdots \\
\Delta u_2(n+M-1)
\end{bmatrix}_{2M \times 1}
\]  

(5.26)
where \( \Delta u_1(n) \) is the move implemented for controller output 1 \( (F_{IN1}) \) and \( \Delta u_2(n) \) is the move implemented for controller output 2 \( (F_{IN2}) \).

The algorithm is formulated here for a TCTILS which has two controller outputs and two process variables. The method can be directly extended to more complex processes.

5.4.1 The Adaptive Strategy

The GSA-DMC strategy builds on design steps as discussed in section 5.4 and the DMC control move calculation by gain scheduler. The adaptive strategy as shown in Figure 5.8 involves designing three linear DMC controllers. Each controller has a model-based optimizer and a model-based predictor. The approach described here involves designing and combining three linear DMC controllers which are designed using designing steps as discussed in section 5.4. However, this technique can involve designing and combining any number of linear DMC controllers.

![Figure 5.8 Schematic of GSA-DMC scheme for TCTILS](image)
As explained above, all linear DMC controllers use the same values for T, P, N and M; while $\gamma_1$, $\gamma_2$, $\lambda_1$ and $\lambda_2$ varies for each linear DMC controller. The three linear DMC controllers each compute their own control action. One of the controller output is selected by gain scheduler based on the value of the current setpoint of each process variable to yield a single set of control moves forwarded to the final control elements. Although three controllers are employed here, the method can be expanded to include as many local linear DMC controllers as the practitioner would like. The use of three linear DMC controllers is the minimum needed to reasonably control a nonlinear process. The performance of the GSA-DMC controller will be better, if more linear DMC controllers are used. There are no theoretical guidelines to illustrate how many linear controllers should be used in the adaptive control strategy to give optimal performance (Yu et al 1992). While this method will often not capture the severe nonlinear behaviors associated with many processes, it will provide significant benefit over the non-adaptive DMC controller.

5.5 DESIGN OF NON LINEAR DMC

In the GSA-DMC scheme, when the setpoint is changed from one operating regime to another operating regime, the respective linear DMC controllers are also change abruptly and will reduce the performance of the process. This drawback can be eliminated by schemes like Nonlinear DMC (NDMC) scheme or Fuzzy gain scheduling DMC scheme. In the present work, NDMC scheme is used.

5.5.1 Takagi-Sugino Fuzzy Inference System

A simple way to describe a nonlinear dynamic system using multiple linear models has been proposed by Takagi-Sugeno (1985) and it is being used in the present work to develop NDMC scheme. The proposed
control scheme consists of a family of local linear controllers and a scheduler. As suggested by Kuipers and Astrom (1994), either local controller outputs or the local controller parameters can be interpolated. In the case of interpolation of controller parameters, the controller’s structure have to be assumed as homogeneous, whereas interpolation of controllers output does not impose any such constraints. In the present work, a method of interpolation of local controller outputs is used. In the present work, at each sampling instant, the scheduler will assign weights for each linear DMC and the average weighted sum of the outputs will be applied as a control input to the TCTILS. In the present work, operating regime approach (Aufderheide and Wayne Bequette 2003; Tian and Hoo 2005) is used to develop NDMC scheme. Since global information can be applied to determine the control input at each sampling instant, the nonlinear model based controller is expected to achieve better control performance. Recently, stability analysis of a multi-model predictive control algorithm with an application to the control of chemical reactors has been reported by Özkan Leyla and Kothare (2006).

Takagi and Sugeno (1985) proposed models where the consequent part of the rule is described by a linear regression model. These models are easier to identify because each rule describes a fuzzy regime in which the output depends on the inputs in a linear manner. An example of such a model is given in Equation (5.27):

\[ R^1: \text{IF } x \text{ is } A_1 \text{ THEN } y = a_1 x + b_1 \]
\[ R^2: \text{IF } x \text{ is } A_2 \text{ THEN } y = a_2 x + b_2 \]
\[ R^3: \text{IF } x \text{ is } A_3 \text{ THEN } y = a_3 x + b_3 \]  
(5.27)

or in general format:

\[ R^i: \text{IF } x \text{ is } A_i \text{ THEN } y_i = a_i^T x + b_i, \quad i=1,2,\ldots, N \]  
(5.28)
The model as given in Equation (5.28) can represent multi-input, multi-output static and dynamic systems. The global output of the system \( y \) can be calculated using the outputs of the individual consequents \( y_i \), using the weighted mean formula:

\[
y = \frac{\sum_{i=1}^{N} \mu_{A_i}(x) y_i}{\sum_{i=1}^{N} \mu_{A_i}(x)} \tag{5.29}
\]

where \( N \) is the number of rules in the rule base. \( \mu_{A_i} \) is the degree of membership of the antecedent of rule \( i \). If the antecedent is a multidimensional proposition, such as given in Equation (5.30):

\[
\text{IF } x_1 \text{ is } A_{i,1} \text{ AND } x_2 \text{ is } A_{i,2} \text{ AND } ... \text{ THEN } y_i = a_i^T x + b_i \tag{5.30}
\]

then the degree of membership is calculated as:

\[
\mu_{A_i}(x) = \mu_{A_i}(x_1) \land \mu_{A_i}(x_2) \land ... \tag{5.31}
\]

where \( \land \) is the minimum operator.

The schematic diagram of NDMC scheme for TCTILS is shown in Figure 5.9. The NDMC scheme consists of three linear DMCs which are same as those used in GSA-DMC scheme and T-S FIS as scheduler. The scheduler will assign weights for each linear DMC and the average weighted outputs will be applied as control inputs to the TCTILS. In Figures 5.10 and 5.11, linearised operating regimes of TCTILS with three T-S rules and bell shaped membership functions (MFs) for T-S FIS1 and T-S FIS2 are shown constituting a local linear description of the functional relationship between output weight \( w \) and setpoint \( x \). In NDMC, \( w_{1i} \) and \( w_{2i} \) (\( I = 1, 2 \) and 3) are weight outputs of T-S FIS1 and T-S FIS2 respectively. Similarly, \( x_1 \) (setpoint SP1) and \( x_2 \) (setpoint SP2) are represent inputs to T-S FIS1 and T-S FIS2 respectively.
Figure 5.9 Schematic of NDMC scheme for TCTILS

R^1: IF x_1 is h_{11} THEN w_{11} = a_{11}x_1 + b_{11}
R^2: IF x_1 is h_{12} THEN w_{12} = a_{12}x_1 + b_{12}
R^3: IF x_1 is h_{13} THEN w_{13} = a_{13}x_1 + b_{13}

Figure 5.10 Local linear description using T-S FIS1 (a) Linearised operating regimes of h_1 (b) Bell shaped MFs (c) T-S rules
From the slope of each linearised regime (represented as dashed line) following linear models can be derived.

For Linearised regimes of TANK1

\[ w_{11} = 18.15x_1 + 0 \]
\[ w_{12} = 10.4243x_1 + 70.1 \]
\[ w_{13} = 6.6763x_1 + 166.187 \]

For Linearised regimes of TANK2

\[ w_{21} = 18.15x_2 + 0 \]
\[ w_{22} = 10.4243x_2 + 70.1 \]
\[ w_{23} = 6.6763x_2 + 166.187 \]

Figure 5.11 Local linear description using T-S FIS2

(a) Linearised operating regimes of \( h_2 \)
(b) Bell shaped MFs
(c) T-S rules
The generalized bell shaped MFs of the following form are used for both T-S FIS1 and T-S FIS2.

\[
\mu(x) = \frac{1}{1 + \left| \frac{x - c}{a} \right|^{2b}} \quad (5.33)
\]

where the parameters \(a\) and \(b\) vary the width of the curve and the parameter \(c\) locates the center of the curve. The parameter \(b\) should be positive. The parameters used for T-S FIS1 and T-S FIS2 are presented in Table 5.6.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>T-S FIS1</th>
<th>T-S FIS2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuzzy Inference System</td>
<td>Sugeno</td>
<td>Sugeno</td>
</tr>
<tr>
<td>Input</td>
<td>Setpoint (h_1)</td>
<td>Setpoint (h_2)</td>
</tr>
<tr>
<td>Output</td>
<td>Controller weights (w_{11}, w_{12}, w_{13})</td>
<td>Controller weights (w_{21}, w_{22}, w_{23})</td>
</tr>
<tr>
<td>Universe of discourse – input</td>
<td>0 to 50</td>
<td>0 to 50</td>
</tr>
<tr>
<td>Universe of discourse – output</td>
<td>0 to 1</td>
<td>0 to 1</td>
</tr>
<tr>
<td>No. of MFs</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Type of MFs</td>
<td>Generalized Bell</td>
<td>Generalized Bell</td>
</tr>
<tr>
<td>Linguistic variables</td>
<td>(h_{11}, h_{12}, h_{13})</td>
<td>(h_{21}, h_{22}, h_{23})</td>
</tr>
<tr>
<td>Defuzzification method</td>
<td>Weighted average</td>
<td>Weighted average</td>
</tr>
</tbody>
</table>

In the present work, global controller output is estimated as given by the Equations (5.34) and is applied as control signal to control TCTILS.

\[
\begin{align*}
    u_1 &= (w_{11}u_{11} + w_{12}u_{21} + w_{13}u_{31}) / (w_{11} + w_{12} + w_{13}) \\
    u_2 &= (w_{21}u_{12} + w_{22}u_{22} + w_{23}u_{32}) / (w_{21} + w_{22} + w_{23}) \\
\end{align*}
\]  

(5.34)

where

\(u_1, u_2\) – global outputs of NDMC

\(w_{11}, w_{12}, w_{13}\) – weight outputs of T-S FIS1
$w_{21}, w_{22}, w_{23}$ – weight outputs of T-S FIS2
$u_{11}, u_{12}$ – outputs from DMC1
$u_{21}, u_{22}$ – outputs from DMC2
$u_{31}, u_{32}$ – outputs from DMC3

The accuracy of the global outputs of NDMC is increased by tuning parameters of generalized bell MFs optimally using real coded GA. The sequence of steps involved in tuning of parameters of generalized bell MFs optimally using real coded GA is similar to steps as shown in Figure 4.4. The performance index used in this tuning process is to minimize average ISE of $h_1$ and $h_2$ simultaneously. The list of decision variables and parameters of real coded GA for optimal tuning of T-S FIS1 and T-S FIS2 are presented in the Table 5.7.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>No of decision variables</td>
<td>18</td>
</tr>
<tr>
<td>- $a$, $b$ and $c$ parameters of generalised bell MFs $h_{11}$, $h_{12}$ and $h_{13}$ of T-S FIS1</td>
<td></td>
</tr>
<tr>
<td>- $a$, $b$ and $c$ parameters of generalised bell MFs $h_{21}$, $h_{22}$ and $h_{23}$ of T-S FIS2</td>
<td></td>
</tr>
<tr>
<td>Precision of parameters</td>
<td>0.001</td>
</tr>
<tr>
<td>Population size</td>
<td>20</td>
</tr>
<tr>
<td>Termination criterion</td>
<td>20 epochs</td>
</tr>
<tr>
<td>Crossover probability</td>
<td>0.85</td>
</tr>
<tr>
<td>Mutation probability</td>
<td>0.15</td>
</tr>
<tr>
<td>Selection method</td>
<td>Roulette wheel</td>
</tr>
<tr>
<td>Crossover method</td>
<td>Arithmetic crossover</td>
</tr>
<tr>
<td>Mutation method</td>
<td>Non-uniform mutation</td>
</tr>
<tr>
<td>Objective function</td>
<td>$J = \frac{1}{2} \left[ \int_{t=0}^{t} \varepsilon_1^2 dt + \int_{t=0}^{t} \varepsilon_2^2 dt \right]$.</td>
</tr>
</tbody>
</table>

The optimally tuned parameters of bell MFs for T-S FIS1 and T-S FIS2 are presented in Table 5.8 and tuned bell MFs are shown in Figure 5.12.
Table 5.8  Real coded GA tuned parameters of generalized bell MFs for T-S FIS1 and T-S FIS2

<table>
<thead>
<tr>
<th>Linguistic variable</th>
<th>Parameter name</th>
<th>Tuned Parameter value</th>
<th>Linguistic variable</th>
<th>Parameter name</th>
<th>Tuned Parameter value</th>
</tr>
</thead>
<tbody>
<tr>
<td>h11</td>
<td>a</td>
<td>4.162</td>
<td>h21</td>
<td>a</td>
<td>4.162</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>1.954</td>
<td></td>
<td>b</td>
<td>2.954</td>
</tr>
<tr>
<td></td>
<td>c</td>
<td>1.041</td>
<td></td>
<td>c</td>
<td>1.041</td>
</tr>
<tr>
<td>h12</td>
<td>a</td>
<td>8.975</td>
<td>h22</td>
<td>a</td>
<td>9.012</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>3.499</td>
<td></td>
<td>b</td>
<td>2.131</td>
</tr>
<tr>
<td></td>
<td>c</td>
<td>13.021</td>
<td></td>
<td>c</td>
<td>13.048</td>
</tr>
<tr>
<td>h13</td>
<td>a</td>
<td>19.997</td>
<td>h23</td>
<td>a</td>
<td>24.898</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>3.489</td>
<td></td>
<td>b</td>
<td>3.489</td>
</tr>
<tr>
<td></td>
<td>c</td>
<td>50.000</td>
<td></td>
<td>c</td>
<td>50.000</td>
</tr>
</tbody>
</table>

Figure 5.12  Real coded GA tuned MFs of (a) T-S FIS1 (b) T-S FIS2

The evolution of performance index $J$ value during various epochs in real coded GA based tuning process of T-S FIS1 is shown in Figure 5.13. The performance index $J$ is converged in $8^{th}$ generation. In this tuning process, performance index $J$ which is an average ISE of $h_1$ and $h_2$ is reduced to 486.4.
Figure 5.13 Evolution of performance index $J$ during tuning of NDMC using real coded GA

The evolution of parameters of bell MFs during various epochs in real coded GA based tuning process of T-S FIS1 is shown in Figure 5.14.

Figure 5.14 Evolution of various parameters during tuning of MFs of T-S FIS1 using real coded GA  

a) $a$ of $h_{11}$,  b) $b$ of $h_{11}$,  
c) $c$ of $h_{11}$,  d) $a$ of $h_{12}$,  e) $b$ of $h_{12}$,  
f) $c$ of $h_{12}$,  g) $a$ of $h_{13}$,  h) $b$ of $h_{13}$ and  i) $c$ of $h_{13}$
Figure 5.14 (Continued)
Figure 5.14 (Continued)
5.6 SIMULATION STUDIES

In all the simulation runs, the TCTILS is simulated using the nonlinear first principle model (Equations (2.12) and (2.13)). The output variables are computed by solving the nonlinear differential equations using differential equation solver in MATLAB R2009a.

5.6.1 Servo Performance

The setpoint variations as shown in Figure 5.15(a) are introduced for assessing the tracking capability of the GSA-DMC and NDMC schemes. The variation in process outputs and controller outputs are shown in Figure 5.15 (a) and Figure 5.15(b) respectively. A portion of the process outputs are shown in Figure 5.15(c). From the response, it can be inferred that, the NDMC scheme is able to maintain the tank levels $h_1$ and $h_2$ at the respective setpoints better than GSA-DMC scheme. The NDMC scheme produces response with faster rise time, less interaction and settles to the setpoint faster compared to GSA-DMC scheme.

![Figure 5.15 Servo response of TCTILS with GSA-DMC and NDMC schemes](image)

(a)

**Figure 5.15** Servo response of TCTILS with GSA-DMC and NDMC schemes (a) Process output (b) Controller output (c) Portion of Process output
The values of performance indices for GSA-DMC and NDMC schemes are computed and presented in Table 5.9. From Table 5.9, it is inferred that the values of performance indices for NDMC scheme are found to be considerably less than GSA-DMC scheme.

Figure 5.15 (Continued)
Table 5.9 Comparison of performance indices of GSA-DMC and NDMC schemes for setpoint tracking

<table>
<thead>
<tr>
<th>Level</th>
<th>Performance Indices</th>
<th>NDMC</th>
<th>GSA-DMC</th>
</tr>
</thead>
<tbody>
<tr>
<td>h₁</td>
<td>ISE</td>
<td>404.9</td>
<td>766.5</td>
</tr>
<tr>
<td></td>
<td>IAE</td>
<td>150.5</td>
<td>252.7</td>
</tr>
<tr>
<td></td>
<td>ITAE</td>
<td>1.3579e+5</td>
<td>1.8008e+5</td>
</tr>
<tr>
<td>h₂</td>
<td>ISE</td>
<td>557.5</td>
<td>799.9</td>
</tr>
<tr>
<td></td>
<td>IAE</td>
<td>165.1</td>
<td>217.9</td>
</tr>
<tr>
<td></td>
<td>ITAE</td>
<td>1.258e+5</td>
<td>1.5096e+5</td>
</tr>
</tbody>
</table>

The time domain specifications such as rise time, settling time and % of overshoot for the response produced by GSA-DMC and NDMC schemes are calculated and presented in Table 5.10. From Table 5.10, it can be inferred that NDMC scheme produces response with faster rise time, less interaction and settles to the setpoint faster compared to GSA-DMC scheme.

Table 5.10 Comparison of time domain performance of NDMC and GSA-DMC schemes for setpoint tracking

<table>
<thead>
<tr>
<th>Step response</th>
<th>Control schemes</th>
<th>Rise time (second)</th>
<th>Settling time (second)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>h₁</td>
<td>NDMC</td>
<td>5</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>GSA-DMC</td>
<td>8</td>
<td>11</td>
</tr>
<tr>
<td>h₂</td>
<td>NDMC</td>
<td>7</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>GSA-DMC</td>
<td>12</td>
<td>18</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Interaction effect</th>
<th>Control schemes</th>
<th>% of overshoot</th>
<th>Settling time (second)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NDMC</td>
<td>1.61 %</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>GSA-DMC</td>
<td>2.46 %</td>
<td>15</td>
</tr>
</tbody>
</table>
The observations (both qualitative and quantitative) of the above simulation study are as follows:

Both the GSA-DMC and NDMC schemes are able to maintain the \( h_1 \) and \( h_2 \) at the respective setpoints. However, the performances of NDMC scheme at all the operating points are found to be better than GSA-DMC scheme, as faster rise time, less interaction and settles to the setpoint faster.

5.6.2 Servo with Regulatory Performance

Simulation studies are carried out to demonstrate the disturbance rejection capability of the NDMC and GSA-DMC schemes at nominal and at shifted operated points.

5.6.2.1 Performance Study for Step Disturbance \( F_{IN1d} \)

A step disturbance \( F_{IN1d} \) of magnitude 25 LPH is introduced as disturbance to TANK1 at a simulation time of 800\(^{th}\) second and the value is maintained upto simulation time of 1499\(^{th}\) second and is then removed at simulation time of 1500\(^{th}\) second. The variation in process outputs and controller outputs for GSA-DMC and NDMC schemes are shown in Figure 5.16 (a) and 5.16(b) respectively.
Figure 5.16 Servo with regulatory response of TCTILS with NDMC and GSA-DMC schemes in the presence of setpoint change and load change in TANK1  

(a) Process output  (b) Controller output

The following observations can be drawn from the simulation studies:

- In the simulation time period from 800\textsuperscript{th} second to 999\textsuperscript{th} second of Figure 5.16(a), it can be inferred that NDMC scheme is able to reject the disturbance quickly and bring the levels $h_1$ and $h_2$ back to the nominal value of the respective setpoints better than GSA-DMC scheme. This part of the simulation demonstrates that the NDMC scheme is able to reject the disturbance at the nominal operating point better than GSA-DMC scheme.
With the disturbance being persistent, a step change in the setpoint $h_1$ and setpoint $h_2$ are introduced at simulation time of $1000^{th}$ second and simulation time of $1250^{th}$ second respectively. It can be noted that the NDMC scheme is able to maintain the levels $h_1$ and $h_2$ at the respective setpoints better than GSA-DMC scheme, as evident from simulation time period of $1000^{th}$ second to $1499^{th}$ second of Figure 5.16(a).

At simulation time of $1500^{th}$ second, both step change in the setpoint $h_1$ as well as a step disturbance $F_{IN1d}$ are removed simultaneously. It can be inferred from the response that the performance of NDMC scheme is found to be better than GSA-DMC scheme. This part of the simulation demonstrates that the NDMC scheme is able to reject the disturbance as well as maintain the process variables at the respective setpoints better than GSA-DMC scheme.

It should be noted from Figure 5.16 that the overall performance of the NDMC scheme is found to be better than GSA-DMC scheme.

The performance indices are computed for GSA-DMC and NDMC schemes and are presented in Table 5.11.

**Table 5.11 Comparison of performance indices of GSA-DMC and NDMC schemes in the presence of setpoint change and load change in TANK1**

<table>
<thead>
<tr>
<th>Level</th>
<th>Performance Indices</th>
<th>NDMC</th>
<th>GSA-DMC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ISE</td>
<td>588.8</td>
<td>853.6</td>
</tr>
<tr>
<td>$h_1$</td>
<td>IAE</td>
<td>130.2</td>
<td>187.6</td>
</tr>
<tr>
<td></td>
<td>ITAE</td>
<td>1.0075e+5</td>
<td>1.0662e+5</td>
</tr>
<tr>
<td></td>
<td>ISE</td>
<td>845.6</td>
<td>992.5</td>
</tr>
<tr>
<td>$h_2$</td>
<td>IAE</td>
<td>153.9</td>
<td>185.7</td>
</tr>
<tr>
<td></td>
<td>ITAE</td>
<td>1.1844e+5</td>
<td>1.2621e+5</td>
</tr>
</tbody>
</table>
The values of performance indices for NDMC scheme are found to be considerably less than that of GSA-DMC scheme for the servo-regulatory performance case also.

5.6.2.2 Performance Study for Step Disturbance $F_{IN2d}$

Similarly, a step disturbance $F_{IN2d}$ of magnitude 25 LPH is introduced as disturbance to TANK2 at a simulation time of 800$^{th}$ second and the value is maintained up to simulation time of 1499$^{th}$ second and is then removed at simulation time of 1500$^{th}$ second. The variation in process outputs and controller outputs are shown in Figure 5.17 (a) and 5.17 (b) respectively.

![Figure 5.17](image)

Figure 5.17 Servo with regulatory response of TCTILS with NDMC and GSA-DMC schemes in the presence of setpoint change and load change in TANK2 (a) Process output (b) Controller output
The observations drawn from this simulation studies are similar to that of the performance study for step disturbance in $F_{IN1d}$ as discussed in section 5.6.2.1. It is observed from Figure 5.17 that the performance of the NDMC scheme is found to be better than GSA-DMC scheme.

The values of performance indices are computed for GSA-DMC and NDMC schemes and are presented in Table 5.12. The values of performance indices for NDMC scheme are found to be considerably less than that of GSA-DMC scheme for the performance of step disturbance in $F_{IN2d}$ case also.

Table 5.12 Comparison of performance indices of GSA-DMC and NDMC schemes in the presence of setpoint change and load change in TANK2

<table>
<thead>
<tr>
<th>Level</th>
<th>Performance Indices</th>
<th>NDMC</th>
<th>GSA-DMC</th>
</tr>
</thead>
<tbody>
<tr>
<td>h₁</td>
<td>ISE</td>
<td>538.7</td>
<td>836.3</td>
</tr>
<tr>
<td></td>
<td>IAE</td>
<td>106.9</td>
<td>180.5</td>
</tr>
<tr>
<td></td>
<td>ITAE</td>
<td>9.1809e+4</td>
<td>1.089e+5</td>
</tr>
<tr>
<td>h₂</td>
<td>ISE</td>
<td>769.3</td>
<td>915.3</td>
</tr>
<tr>
<td></td>
<td>IAE</td>
<td>138.6</td>
<td>167.7</td>
</tr>
<tr>
<td></td>
<td>ITAE</td>
<td>9.4027e+4</td>
<td>9.5073e+4</td>
</tr>
</tbody>
</table>

From the simulation responses (qualitative analysis) and performance indices values (quantitative analysis), it is understood that NDMC scheme rejects disturbance better than GSA-DMC scheme.

5.6.3 Robust Performance

Simulation studies are carried out to demonstrate the robustness of the GSA-DMC and NDMC schemes at nominal and at shifted TCTILS parameter. The robustness of GSA-DMC and NDMC schemes are tested by
varying the valve coefficient $\beta_{12}$ of $MV_{12}$ from its nominal value (35cm$^3$/s). The increase in $\beta_{12}$ physically represents the increase in interaction between two tanks. Since, time constant and process gain of TCTILS are depend on $\beta_{12}$, the change in $\beta_{12}$ will change time constant and process gain. The robustness of the GSA-DMC and NDMC schemes are tested by increasing the $\beta_{12}$ value from 0% (no change in $\beta_{12}$) to 50% in steps of 10% change. The variation in process outputs and control outputs of robustness performance of GSA-DMC and NDMC schemes for 0 %, 30 % and 50 % changes in $\beta_{12}$ are shown in the Figure 5.18. From the Figure 5.18(a), it is observed that the

![Figure 5.18](image-url)
NDMC scheme is able to maintain the $h_1$ and $h_2$ at the respective setpoints better than GSA-DMC scheme even though the process parameter $\beta_{12}$ is increased up to 50% from its nominal value. It is also observed that the NDMC scheme produces response with lesser overshoot and settles to the setpoint faster. From Figure 5.18(b), it is understood that NDMC scheme consumes lesser control effort and produces control output with lesser oscillation compared to GSA-DMC scheme.

The performance indices of GSA-DMC and NDMC schemes are computed and presented in Table 5.13. From Table 5.13, it can be inferred that the values of performance indices for NDMC scheme are found to be considerably less than that for GSA-DMC scheme.

Table 5.13 Robustness performance values of NDMC and GSA-DMC schemes for various increased $\beta_{12}$ values

<table>
<thead>
<tr>
<th>Control Schemes</th>
<th>% change in $\beta_{12}$</th>
<th>$h_1$</th>
<th>$h_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>ISE</td>
<td>IAE</td>
</tr>
<tr>
<td>NDMC</td>
<td>No change</td>
<td>758</td>
<td>122.3</td>
</tr>
<tr>
<td></td>
<td>10 %</td>
<td>757.4</td>
<td>124.4</td>
</tr>
<tr>
<td></td>
<td>20 %</td>
<td>757.6</td>
<td>126.7</td>
</tr>
<tr>
<td></td>
<td>30 %</td>
<td>758.5</td>
<td>129.2</td>
</tr>
<tr>
<td></td>
<td>40 %</td>
<td>760.2</td>
<td>131.7</td>
</tr>
<tr>
<td></td>
<td>50 %</td>
<td>762</td>
<td>134.4</td>
</tr>
<tr>
<td>GSA-DMC</td>
<td>No change</td>
<td>1031</td>
<td>183.2</td>
</tr>
<tr>
<td></td>
<td>10 %</td>
<td>1020.5</td>
<td>184.4</td>
</tr>
<tr>
<td></td>
<td>20 %</td>
<td>1013</td>
<td>185.7</td>
</tr>
<tr>
<td></td>
<td>30 %</td>
<td>1008.2</td>
<td>186.9</td>
</tr>
<tr>
<td></td>
<td>40 %</td>
<td>1006.1</td>
<td>188.3</td>
</tr>
<tr>
<td></td>
<td>50 %</td>
<td>1006.4</td>
<td>189.6</td>
</tr>
</tbody>
</table>
From the simulation responses (qualitative analysis) and performance indices values (quantitative analysis), it is understood that NDMC scheme is more robust than GSA-DMC scheme.

5.6.4 Observation from Simulation Studies

Simulation responses (qualitative analysis) show that NDMC strategy significantly outperforms GSA-DMC scheme, especially when facing with setpoint variations, load disturbances and diversified uncertainty in dynamic interaction. And also the NDMC scheme helps to produce response with less rise time, less interaction and settles to the setpoint faster in the entire operating regime compared to GSA-DMC scheme. From the performance indices values (quantitative analysis), it is observed that the robustness and control performance of NDMC at all operating points are found to be better than GSA-DMC.

The performance of the NDMC remains constant as the dynamic behavior of the process changes by weighting the DMC controller output moves from each model to account for the changing dynamic behavior of the process. In effect, the NDMC strategy is changing the amount of control effort needed as the process dynamics are changed.

The merits of NDMC scheme are summarized as follows: (1) it can make use of the traditional linear controller design; (2) it helps to produce response with quicker rise time, less interaction effect and settles to the setpoint faster in entire operating regime compared to traditional GSA-DMC scheme; (3) the implementation of NDMC scheme is quite straightforward
and easy. With its significant and simple controller structure, and excellent and robust performance, the NDMC scheme presents to be a promising approach to nonlinear process control, alternative to existing GSA-DMC scheme in the literature.

5.7 COMPARISON OF ADD-PI CONTROL, AMFC AND NDMC SCHEMES

From six control schemes (NADD-PI, ADD-PI, ATFC, AMFC, GSA-DMC and NDMC) developed for TCTILS using MMAC strategy, the best three control schemes such as ADD-PI control scheme, AMFC scheme and NDMC scheme are selected from respective broad class such as conventional control, intelligent control and predictive control.

The setpoint variations as shown in Figure 5.19(a) are introduced for assessing the tracking capability of the ADD-PI control, AMFC and NDMC schemes. The variation in process outputs and control outputs of ADD-PI, AMFC and NDMC schemes are shown in the Figure 5.19. From the response, it can be inferred that, the NDMC scheme is able to maintain the tank levels $h_1$ and $h_2$ at the respective setpoints better than ADD-PI control and AMFC schemes. The NDMC scheme produces response with faster rise time, less interaction and settles to the setpoint faster compared to ADD-PI control and AMFC schemes. From Figure 5.19(b), it is inferred in the case of NDMC scheme that the change in controller output is smooth even sudden setpoint changes occur. Whereas control kicks occur in the case of ADD-PI control and AMFC schemes.
The values of performance indices for ADD-PI control, AMFC and NDMC schemes are presented in Table 5.14. From Table 5.14, it can be inferred that the values of performance indices for NDMC scheme are found to be considerably less than that for ADD-PI control and AMFC schemes.
### Table 5.14 Comparison of performance indices of ADD-PI, AMFC and NDMC schemes for setpoint tracking

<table>
<thead>
<tr>
<th>Level</th>
<th>Performance Indices</th>
<th>ADD-PI CONTROL SCHEME</th>
<th>AMFC SCHEME</th>
<th>NDMC SCHEME</th>
</tr>
</thead>
<tbody>
<tr>
<td>h₁</td>
<td>ISE</td>
<td>204.8246</td>
<td>127.1170</td>
<td>26.8446</td>
</tr>
<tr>
<td></td>
<td>IAE</td>
<td>73.9027</td>
<td>41.9384</td>
<td>29.5113</td>
</tr>
<tr>
<td></td>
<td>ITAE</td>
<td>2.0254e+04</td>
<td>9.8868e+03</td>
<td>7.2862e+03</td>
</tr>
<tr>
<td>h₂</td>
<td>ISE</td>
<td>651.1254</td>
<td>456.5034</td>
<td>21.7103</td>
</tr>
<tr>
<td></td>
<td>IAE</td>
<td>131.8274</td>
<td>110.7478</td>
<td>27.1921</td>
</tr>
<tr>
<td></td>
<td>ITAE</td>
<td>3.4854e+04</td>
<td>2.1715e+04</td>
<td>6.6613e+03</td>
</tr>
</tbody>
</table>

The time domain specifications such as rise time, settling time and % of overshoot for the response produced by ADD-PI, AMFC and NDMC schemes are calculated and presented in Table 5.15. From Table 5.15, it can be inferred that NDMC scheme produces response with faster rise time, less interaction and settles to the setpoint faster compared to ADD-PI control and AMFC schemes.

### Table 5.15 Comparison of time domain performance of ADD-PI, AMFC and NDMC schemes for setpoint tracking

<table>
<thead>
<tr>
<th>Step response</th>
<th>Level</th>
<th>Control schemes</th>
<th>Rise time (second)</th>
<th>Settling time (second)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>h₁</td>
<td>ADD-PI</td>
<td>11</td>
<td>32</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AMFC</td>
<td>9</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NDMC</td>
<td>5</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>h₂</td>
<td>ADD-PI</td>
<td>12</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AMFC</td>
<td>9</td>
<td>21</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NDMC</td>
<td>7</td>
<td>14</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Interaction effect</th>
<th>Control schemes</th>
<th>% of overshoot</th>
<th>Settling time (second)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADD-PI</td>
<td>4.1 %</td>
<td>26</td>
<td></td>
</tr>
<tr>
<td>AMFC</td>
<td>3.6 %</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>NDMC</td>
<td>1.61 %</td>
<td>6</td>
<td></td>
</tr>
</tbody>
</table>
The observations (both qualitative and quantitative) of the above simulation study are as follows:

All the three control schemes are able to maintain the $h_1$ and $h_2$ at the respective setpoints. However, the performances of NDMC scheme at all the operating points are found to be better than ADD-PI control and AMFC schemes, as faster rise time, less interaction and settles to the setpoint faster.

5.8 SUMMARY

In this chapter, a simple and straightforward procedure is suggested for designing a NDMC scheme and GSA-DMC scheme using MMAC strategy for the TCTILS, which exhibits significant variation in process parameters as the operating point changes and also exhibits nonlinear coupling dynamics. This work develops an adaptive strategy that builds upon linear controller design methods for creating a robust MMAC strategy for DMC. Firstly, MMAC based DMC scheme using linear models and gain scheduler is designed for TCTILS. Even though gain scheduling based adaptive control scheme is most widely used method in industries, when the setpoint is changed from one operating regime to another operating regime, the respective linear DMC is also switched abruptly which may cause sudden kicks and hence may reduce the performance of the process. Secondly, nonlinear DMC scheme is designed using T-S FIS as scheduler and linear DMCs. The scheduler will assign weights for each linear DMC and the average weighted sum of the outputs will be applied as global control input to the TCTILS. The NDMC scheme eliminates the drawback of GSA-DMC scheme and improves the overall performance of TCTILS. An effective DMC controller’s parameter tuning algorithm is devised to tune parameters of linear DMCs in GSA-DMC and membership functions of T-S FIS in NDMC schemes using real coded GA. The contributions of the method presented here include an NDMC strategy that:
is straightforward to implement and use,
relies on the linear control knowledge of plant personnel, and
is reliable for a broad class of process applications.

Also, simulation studies are carried out to compare the servo capability of the ADD-PI control scheme, AMFC scheme and NDMC scheme. From the simulation response and performance indices values, it can be inferred that, the NDMC scheme is able to maintain the tank levels $h_1$ and $h_2$ at the respective setpoints better than ADD-PI control and AMFC schemes. And also NDMC scheme produces response with faster rise time, less interaction and settles to the setpoint faster than other control schemes.

The real time implementation of NDMC and GSA-DMC schemes in TCTILS experimental setup are discussed in detail in the next chapter.