The Modular Multivariable Controller: I: Steady-State Properties

T. A. Meadowcroft and G. Stephanopoulos
Laboratory for Intelligent Systems in Process Engineering, Dept. of Chemical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139

C. Brosilow
Dept. of Chemical Engineering, Case-Western Reserve University, Cleveland, OH 44106

The modular multivariable controller (MMC) represents a multivariable controller design methodology which is based on the solution of multiobjective optimization problems using the strategy of lexicographic goal programming; priority-driven, sequential satisfaction of objectives. This article formally introduces the concept of the MMC, analyzes its static characteristics, and proposes a specific methodology for the design of steady-state MMCs. It is shown that the framework of MMC can explicitly handle all types of control objectives (for example, equality or inequality specifications on controlled outputs), and constraints on manipulations. Its priority-driven, sequential satisfaction of control objectives leads to a modular, hierarchical structure of controllers with specific objectives. The modular character of MMC allows the explicit maintenance, tuning, and reconfiguration of multivariable control systems, while its hierarchical structure explicitly expresses engineering decisions and trade-offs. Its static design incorporates uncertainty in process gains and automatic reconfiguration to account for failure in sensors and/or actuators. The design of an MMC for a heavy oil fractionator is presented to illustrate the controller's character and the proposed methodology for the design of static MMCs.

Introduction

The design of multivariable process control systems is essentially a multiobjective design problem. The list of objectives reflects the need to achieve, in the presence of external "disturbances," various operational goals (Prett and Garcia, 1988) associated with the economic performance of the process, specifications on production levels and product quality, equipment safety, environmental regulations, and human preferences describing the oscillations or jaggedness of operating trends that the operators will not tolerate. These practical requirements are normally converted into mathematical expressions, stated either as *objectives* or *constraints*. The objectives are functions of variables (Prett and Garcia, 1988), which must be satisfied dynamically in some optimal fashion, for example, minimize steam requirements in the reboiler of a distillation column, maintain the minimum integral square error of a variable from its desired set point. Constraints, on the other hand, are functions of variables which must be maintained within specific bounds, for example, restrictions on the capacity of actuators or the rate of change of manipulated variables, upper or lower bounds on allowable pressures, temperatures, levels, and so on. The constraints are labeled *hard* if no dynamic violations of the bounds is allowed, and soft if temporary violations are allowed in order to satisfy other objectives.

Furthermore, the design of a multivariable controller should address directly a series of additional issues, such as uncertainty in the assumed process model; a very critical issue which has received the focused attention of a lot of work in the last ten years. Another issue is tolerance to faults of control system components (for example, sensors, actuators). Finally, the structure and functionality of a multivariable controller should be transparent to the process engineers and operators, so that they can tune the controllers and maintain their design in the presence of changes in process behavior, or modifications in the operational priorities or revamps in the design of the processing system itself. Such requirements of "transparency" (with a few exceptions, for example, Prett and Garcia, 1988; Campo et al., 1990) are not normally considered as important elements of the design problem by the process controller design literature, at large, despite the obvious practical ramifications. One of the reasons is that the term "transparent design" is rather vague and susceptible to arbitrary handling, informal mathematical treatment and abuse.

To address the multiobjective character of the controller design problem, most of the previous literature has relied on the construction of a single objective utility function; the most typical among the various alternatives being the weighted sum of the multiple objectives (Prett and Garcia, 1988; Morari and Zafiriou, 1989; Prett et al., 1990). The weights are normally determined by an interplay of subjective design judgements (that is, what is more important) and numerical simulations on the dynamic behavior of the process to be controlled. Thus, the resulting controller design problem is usually stated as a minimization problem (static or dynamic, linear or not), with a single objective (weighted sum of objectives) and a set of constraints (see papers in Prett et al., 1990). To account for the impact of modeling errors, an inner optimization problem can be formulated to produce the "worst case" behavior of the controlled system in the presence of potential modelling errors. This min-max formulation of the controller design problem (Morari and Zafiriou, 1989; Cuthrell et al., 1990; Biegler, 1990; Holt and Lu, 1990; Manousiouthakis, 1990; Campo et al., 1990) tends to result in very conservative controllers, which nevertheless do produce very robust closed-loop performance for rather slow processes with large inertia. Zafiriou (1990) has pointed out that the min-max controller design methodology has several weaknesses such as the numerical computations which are too time consuming to be carried out on-line at every sample point and the open-loop character of the minimization problem that does not take into account the feedback from an uncertain plant that exists in reality, thus leading to performance deterioration and instability. Alternatively, instead of augmenting the objective function with the inner maximization problem (in the min-max formulation), to account for robustness to model uncertainty, Zafiriou has tackled the nonlinear nature of the problem directly. Finally, a third approach is to carry out extensive off-line simulations to measure the effect of possible errors on performance and stability. This approach relies on the control system designer to examine all relevant combinations operating regimes and plant uncertainties that could arise. This approach can be very time-consuming if it is carried out thoroughly, and dangerous if it is not.

The single objective approach to the design of model predictive controllers has provided a very appropriate framework for the theoretically sound analysis of control systems. It has also led to a series of successful industrial applications. But, by its own nature it cannot elucidate the most important design decisions, all of which involve the settlement of design trade-offs among conflicting objectives and the demarcation of certain constraints as flexible (as opposed to rigid).

In this article we will develop an alternative approach to the design of model predictive controllers, based on the explicit treatment of the multiobjective character of the design problem. A multiobjective methodology has been proposed (Kreisselmeier and Steinhauser, 1979, 1983) which, with a human designer's interaction, allows a control engineer to optimize the parameters of a conventional controller design in such a way that the trade-offs of his design choices with respect to the multiple objectives is explicit. This methodology, while allowing multiobjective design, does not result in a controller which operates in a multiobjective manner. The primary value of the multiobjective approach in this article is in the understanding that operators and designers gain of the real problem by forcing the designer to make the statement of the design problem as explicit as possible. It also allows the direct utilization of the available control-theoretical analytic knowledge. In the limit, the controller is "optimal" in achieving the desired objectives under the assumed bound on model uncertainties, while maintaining the stated constraints. The resulting controller, which we call the *modular multivariable controller* (MMC), possesses certain interesting practical features such as its ability to handle explicitly the list of desired operational objectives, modularity allowing straightforward reprioritization of control objectives and constraints, on-line tuning which is direct and physically explainable, and direct maintenance in the presence of process design retrofits. Also, it provides explicit fault-tolerant control in the presence of sensor or actuator failures, and allows the operator to take over manual control of the failed subsystem.

The design methodology leading to an MMC is based on the concept of *lexicographic goal programming*. This approach to solving multiobjective optimization problems will be discussed within the scope of the controller design problem. Although the design approach exemplified by MMC addresses both the static and dynamic characteristics of the problem, in this article we will address in detail the design of MMC under steady-state conditions, leaving the dynamic component for a future publication. First, we present the structure and functionality of a static MMC, while in the next section we address the design issues of MMCs robust performance in the presence of model uncertainties. Next we present the characteristic static properties of an MMC, and finally we illustrate MMC within the scope of Shell's Standard Control Problem.

Controller Design as a Problem of Goal Programming

Multiobjective character of the controller design

Let us consider the general statement of a controller design problem:

Regulation of Outputs at the Desired Set Points. Assume that n outputs, $y_1, y_2, \ldots, y_n$ should be kept at the corresponding desired set points $y_{1,sp}, y_{2,sp}, \ldots, y_{n,sp}$, in the presence of load disturbances or set point changes. In a dynamic environment, we cannot keep an output $y_i(t)$ at $y_{i,sp}$ for all time points. Thus, we can formulate the above design objectives as minimizations of square errors, that is:

$$\underset{\Delta m}{\text{Min}}\ F_i = \sum_{k=1}^{P} [y_i(k) - y_{i,sp}]^2 \quad i \in S \tag{1}$$

where S signifies the set of outputs which would be held at desired set points. The index k signifies the sampling time instance k, while P is the number of output sampling time instances in a predictive control horizon.

Maintain Bounded Behavior of Certain Outputs. In addition to the n outputs that must be kept at desired set points, a controller may need to maintain a set of q additional outputs within a bounded region, that is:

$$y_{j,\mathrm{lo}} \leq y_j(k) \leq y_{j,\mathrm{up}} \quad j \in B \tag{2a}$$

where B signifies the set of outputs which are not kept at desired set points but must be kept in a bounded region. This definition creates rigid or "hard" constraints where no excursion outside the feasible region is allowed. A flexible or "soft" representation of bounded output objectives is:

$$\min_{\Delta m} F_j = \sum_{k=1}^{P} (\mathrm{neg}(y_j(k) - y_{j,\mathrm{lo}}) + \mathrm{pos}(y_j(k) - y_{j,\mathrm{up}}))^2 \quad j \in B \tag{2b}$$

$$\mathrm{pos}(x) \equiv \begin{Bmatrix} x;\ x>0 \\ 0;\ x \leq 0 \end{Bmatrix} \quad \mathrm{neg}(x) \equiv \begin{Bmatrix} 0;\ x \geq 0 \\ x;\ x<0 \end{Bmatrix}$$

Note that if we force F_j to 0, no deviations are permitted and Eq. 2b becomes equivalent to Eq. 2a.

Maintain Control Action within Permissible Bounds. The values of the manipulated variables are restricted by physical limitations of the actuators, that is:

$$m_{i,\mathrm{lo}} \leq m_i(k) \leq m_{i,\mathrm{up}} \quad i = 1, 2, \ldots, p \tag{3}$$

Mechanical limitations and/or human preferences impose restrictions on the rate of change of manipulated variables, or equivalently on the control action at every sampling instant, that is:

$$\Delta m_{i,\mathrm{lo}} \leq \Delta m_i(k) \leq \Delta m_{i,\mathrm{up}} \quad i = 1, 2, \ldots, p \tag{4}$$

Accounting for Process Modelling and Model Uncertainty. Let the process model be represented in the standard discrete linear form described in Prett and Garcia (1988):

$$\begin{bmatrix} \Delta x(k+1) \\ y(k+1) \end{bmatrix} = \begin{bmatrix} \Phi & 0 \\ C\Phi & I \end{bmatrix} \begin{bmatrix} \Delta x(k) \\ y(k) \end{bmatrix} + \begin{bmatrix} \Gamma \\ C\Gamma \end{bmatrix} \Delta m(k) + \begin{bmatrix} w_1(k) \\ w_2(k) \end{bmatrix} \tag{5}$$

where

$$y(k) = [y_i(k)] \quad i \in (S \cup B)$$

$$\Delta m(k) = [\Delta m_j(k)] \quad j = 1, 2, \ldots, p$$

$$x = \text{state vector}$$

$$w_1,\ w_2 = \text{disturbances}$$

To account for model uncertainty, we let the elements of Φ and Γ vary between upper and lower bounds:

$$\Phi_{\mathrm{lo}} \leq \Phi \leq \Phi_{\mathrm{up}}, \quad \Gamma_{\mathrm{lo}} \leq \Gamma \leq \Gamma_{\mathrm{up}} \tag{6}$$

In general, the variations of the elements in Eq. 6 may or may not be correlated depending on the model being used.

The statement of the controller design problem as given by Eqs. 1 through 6 leads to a multiobjective optimization problem, as can be easily seen from the list of minimization problems described by Eq. 1. There are many different solution approaches for a multiobjective optimization problem, each representing a different design philosophy. Therefore, which approach one will choose depends entirely on the subjective judgement of a designer, based on what is known to yield a meaningful evaluation of trade-offs among conflicting objectives, the clarity and explainability of the resulting control structures, the efficiency with which the controller is implemented on-line, the ease of tuning the controller on-line, and the ease with which the controller may be updated in the presence of changes in operational priorities, or in the presence of design retrofits in the process itself.

Utility function approach to solving the multiobjective controller design problem

The most common approach in the past literature has been the minimization of a single objective (utility function), the weighted sum of the squared errors given by Eq. 1, that is *Problem P1*:

$$\min_{\Delta m} F = \sum_{i \in S} W_i F_i = \sum_{i \in S} W_i \sum_{k=1}^{P} (y_i(k) - y_{i\mathrm{sp}})^2 \quad i \in S \tag{P1a}$$

subject to constraints Eqs. 2a and 3 through 4. If flexible constraints of the form Eq. 2b are used, they could be included in the objective function as a (weighted) penalty term:

$$\begin{aligned} \min_{\Delta m} F &= \sum_{i \in S} W_i F_i + \sum_{j \in B} W_j F_j \\ &= \sum_{i \in B} W_i \sum_{k=1}^{P} (y_i(k) - y_{i\mathrm{sp}})^2 + \sum_{j \in B} W_j \sum_{k=1}^{P} (\mathrm{neg}(y_j(k) - y_{j,\mathrm{lo}}) + \mathrm{pos}(y_j(k) - y_{j,\mathrm{up}}))^2 \end{aligned} \tag{P1b}$$

subject to constraints Eqs. 3 through 6.

Problem P1 is solved using quadratic programming. Although from a mathematical point of view this approach is formally sound and rigorous, from an engineering point of view it has several drawbacks:

- The numerical values of weights are set through an informal process, which involves subjective judgements by the designer as to the level of importance of each objective vs. the rest. The setting of the values of the weights is completely external to the design approach exemplified by objective Problem P1 and constraints Eq. 2 through 6.
- Changes in the priorities (relative importance) of operational goals must first be scrutinized by process controller designers, who will set the new values for the weights, thus making the updating of controller designs a task for highly skilled personnel.
- The set of constraints Eq. 2a is handled as a set of rigid constraints, which must be satisfied at all times. In industrial practice, temporary violations of constraints may be allowed for some outputs, especially in the presence of an actuators'

saturation or failure. To allow the "softening" of the constraints, Eq. 2a, the objective function in Eq. P1a is augmented with the weighted deviation of the constraint (that is, P1b). In such a case, the value of the weight is established through an informal process.

- Although the performance of the resulting controller in terms of each subobjective, F_i, is affected by the stipulated weight w_i, the actual value of F_i is also governed by the assumed errors in modeling the response of y_i. The impact of modeling errors on the achievable performance of each subobjective, F_i, is implicit. One cannot explicitly shift the adverse effects of large modeling errors on less important subobjectives F_j.
- The on-line tuning of controllers, derived from the solution of problem P1 in the presence of constraints, Eqs. 2 through 6, is also implicit. No explicit causality can be established between process parameters and controller tuning parameters.
- The design validation problem, that is, the confirmation that the specified performance criteria can be met by some controller, is very hard to untangle and can only be approached through *ad-hoc* simulations.

To overcome the above drawbacks, we propose to solve the controller design problem, Eq. 1 subject to constraints Eqs. 2 through 6, using the framework of *lexicographic goal programming* (Ijiri, 1965; Jääskelainen, 1972; Ignizio, 1976, 1982). It is an alternative, mathematically sound approach to solving multiobjective optimization problems, whose primary advantage is in the understanding that one gains of the real problem, by means of explicit construction and analysis of the multiobjective problem.

Goal programming formulation of the controller design problem

Rather than solve the single objective optimization given by P1 subject to constraints Eq. 2 through 6, we will transform the original multiobjective problem to the lexicographic minimization problem of generalized goal programming. As this article is limited to the solution of the static problem, we will simplify the development of the MMC formalism by only addressing the static case. The dynamic formulation is analogous, and will be described in a forthcoming publication. The multiobjective problem is transformed to the lexicographic minimization as follows:

1. Each of the n set point objectives in Eq. 1 is converted into a *goal* of the form:

$$F_i = |y_i(k) - y_{i,sp}| \leq b_i \quad i \in S \tag{7}$$

with b_i signifying the *aspiration level* of objective F_i, that is, a specific value associated with an acceptable level of achievement of objective F_i.

2. Similarly, each of the q constraints Eq. 2 is also converted into two goals:

$$F_{j,\mathrm{up}} = |\mathrm{pos}[y_j(k) - y_{j,\mathrm{up}}]| \leq b_{j,\mathrm{up}} \quad j \in B \tag{8a}$$

$$F_{j,\mathrm{lo}} = |\mathrm{neg}[y_j(k) - y_{j,\mathrm{lo}}]| \leq b_{j,\mathrm{lo}} \quad j \in B \tag{8b}$$

where $b_{j,\mathrm{up}}$ and $b_{j,\mathrm{lo}}$ represent the corresponding aspiration levels. A rigid constraint would have an aspiration level of 0. A positive value for the aspiration level allows some violation of the constraint, making it flexible.

3. Let d_i, $i = 1, 2, \ldots, n$ be the deviations from aspiration levels of the goals, Eq. 7, that is:

$$d_i = b_i - F_i \quad i \in S \tag{9}$$

and since such deviations may be either negative or positive valued, we let:

$$d_i = -\eta_i + \rho_i \quad i \in S \tag{10}$$

where $\eta_i \cdot \rho_i = 0$ and $\eta_i, \rho_i \geq 0$ for $i \in S$. Variables η_i and ρ_i represent the *underachievement* and *overachievement* of the objective, respectively.

Similarly, we introduce deviation variables for the goals in Eq. 8a and for the goals in Eq. 8b:

$$d_{j,\mathrm{up}} = b_{j,\mathrm{up}} - F_{j,\mathrm{up}} \quad j \in B \tag{11a}$$

$$d_{j,\mathrm{lo}} = b_{j,\mathrm{lo}} - F_{j,\mathrm{lo}} \quad j \in B \tag{11b}$$

$$d_{j,\mathrm{up}} = -\eta_{j,\mathrm{up}} + \rho_{j,\mathrm{up}} \quad j \in B \tag{12a}$$

$$d_{j,\mathrm{lo}} = -\eta_{j,\mathrm{lo}} + \rho_{j,\mathrm{lo}} \quad j \in B \tag{12b}$$

For rigid constraints ρ_j must always be 0 because $b_j = 0$ and $F_j \geq 0$, that is, overachievement of a rigid constraint is not possible.

4. Rank all of the goals of Eqs. 7, 8a and 8b according to their importance. Then form the following lexicographic problem:

Problem P2

$$\min_{\Delta m} a = [a_1, a_2, \ldots, a_r] \tag{13}$$

where $r = n + 2 \cdot q$

subject to the following constraints:

$$F_i - \eta_i + \rho_i = b_i \quad i \in S$$

$$F_{j,\mathrm{lo}} - \eta_{j,\mathrm{lo}} + \rho_{j,\mathrm{lo}} = b_{j,\mathrm{lo}} \quad j \in B$$

$$F_{j,\mathrm{up}} - \eta_{j,\mathrm{up}} + \rho_{j,\mathrm{up}} = b_{j,\mathrm{up}} \quad j \in B$$

$$\eta_i,\ \rho_i,\ \eta_{j,\mathrm{lo}},\ \rho_{j,\mathrm{lo}},\ \eta_{j,\mathrm{up}},\ \rho_{j,\mathrm{up}} \geq 0 \quad i \in S,\ j \in B$$

$$\eta_i \cdot \rho_i = 0,\ \eta_{j,\mathrm{lo}} \cdot \rho_{j,\mathrm{lo}} = 0,\ \eta_{j,\mathrm{up}} \cdot \rho_{j,\mathrm{up}} = 0 \quad i \in S,\ j \in B$$

plus the constraints, Eqs. 3 and 4.

The components of the achievement vector $\boldsymbol{a}$ are usually a linear function of the corresponding underachieving goal deviation variable, for example, $a_k = \eta_k \cdot \rho_k$ is not included because it represents the overachievement of a goal, that is, being closer to the set point or having a harder constraint than was indicated by the aspiration level. The lexicographic minimum of problem P2 is defined as follows:

Lexicographic Minimum. The solution $\boldsymbol{a}^{(1)}$ *is preferred* to the solution $a^{(2)}$ if for the goal a_k at any priority level k ($k=1, 2, \ldots, r$), the following is true:

$$a_k^{(1)} < a_k^{(2)}$$

while for all goals of priority higher than k,

$$a_j^{(1)} = a_j^{(2)} \quad j < k \text{ (higher priority)}$$

The lexicographic minimum, $\boldsymbol{a}^*$, is the solution which is preferred over any other.

The definition of the lexicographic minimum implies that the various goals are satisfied consecutively, starting with the most important and proceeding to the least important in order of reduced priorities, rather than simultaneously, as is the case with formulations based on the minimum sum of squared errors. The solution to the lexicographic minimization problem P2 can be found through the solution of a series of optimization problems of the form (Chankong and Haimes, 1983; Ignizio, 1983):

$$\underset{\Delta m}{\textbf{Min}}\ a_k(\eta_k, \rho_k) \quad k=1, 2, \ldots, r$$

s.t. $a_j = \text{Min}\ a_j(\eta_j, \rho_j)$

(for all goals j of priority higher than k)

plus constraints, Eqs. 3 through 6.

The solution of this problem will be optimal as defined in generalized goal programming. That is to say, the solution will be at the lexicographic minimum; as many of the most important objectives as possible will be minimized as much as is possible. This 'optimal' solution is not, in general, the optimal solution of a linear quadratic utility function problem such as that previously discussed. Neither is the linear quadratic optimum necessarily the lexicographic minimum.

Comments on the goal programming based formulation

At this point we should address various objections and arguments that can be made against lexicographic goal programming as a viable framework for the formulation of process controller design problems:

Issue of Aspiration Levels. It could be argued that the aspiration levels b_i for the objectives (goals) F_i, $i \in S$, are as arbitrary (or enlightened) as the values of the weights in problem P1. In real world problems, the value b_i indicating the cumulative time deviation of a controlled output from its set point during a time period equal to the control horizon can be related to the value of an economic impact. For example, in the heavy oil fractionator of the Shell Standard Control Problem (Prett and Garcia, 1988), the cumulative deviation over time of the top or side draw product end points (compositions) from their corresponding setpoints can have a quantifiable economic impact, which in turn could define a priori an acceptable aspiration level for these two control objectives (goals). On the contrary, the weighted least squares formulation of problem P1 first assumes values for the weights and then through simulation computes the cumulative deviations.

Issue of Pareto Suboptimal Solutions. The solution of Problem P1 is Pareto-optimal, that is, for a given set of weights we cannot improve one objective, for example, F_j, without adversely affecting the value of another objective, F_i, $i \neq j$. This is not, in general, the case with the lexicographic minimum of problem P2, for the arbitrarily selected aspiration levels. First, it should be noted that the Pareto-optimality of the solution of problem P1 is an "elusive" ideal solution to the real world problem, and its value is directly related to the values of the weights. Furthermore, the aspiration levels b_i need not be considered as rigid. If the aspiration levels of some goals can be "overachieved", an explicit and directed "tightening" of the overachieved goals will lead to an overall improvement of the "closed-loop" performance. Such an on-line adaptation of the controller's performance is fairly explicit and is based on fairly direct measures of performance. Similarly, a "loosening" of the aspiration level is needed when the goal is consistently "underachieved". Therefore, a controller based on the solution of problem P2 may not be initially Pareto-optimal. However, after the controller has been implemented for a given process, the aspiration levels may be adjusted on-line to achieve Pareto-optimality.

Issue of Assigning Priority Levels to the Various Goals. It may be argued that the assignment of priority levels to the various goals is a very hard task that defies the skills of the designer. The controller design approach espoused by the MMC is based on the belief that *it is far easier for a designer to subjectively assign distinct priority levels to the various control objectives rather than to "capture' his/her own preferences in a quantitative utility function.* Specifically, if a designer can capture his/her performance into a utility function, then automatically he/she has also assigned priority levels to the various objectives, but the reverse is not necessarily true. A simple criterion in selecting the relative priority of two control objectives (goals) would be the answer to the question: "Given the situation where only one goal can be achieved, which one would you choose?" The only issue arises when two goals are presumed to be at the same priority level k. In such cases, a utility function based on the sum of equally weighted goals is an acceptable composite objective at the priority level k. Quite often though, the inability to assign distinct priority levels to various goals stems from the lack of an in-depth understanding of the control problem, or the changing priority levels of two goals in different operational contexts. Thus, within a given context, for example, large production level changes, objective A may be more important than objective B, while in a different context, for example, changes in the distribution of produced products, the reverse is true. A controller based on lexicographic goal programming can accommodate such structural variations in an explicit manner.

In the previous paragraphs we have tried to address the most central differences between the design approaches represented by the formulations of problem P1 and problem P2, and to show that the lexicographic goal programming approach offers a more realistic controller design framework, without sacrificing mathematical rigor. A series of additional issues still need to be addressed, such as: the treatment of model uncertainty and the design of robust controllers, the selection of manipulated variables to carry out the lexicographic minimization of problem P2, the validation of the design problem and others. In this article we will address these issues within the scope of a static MMC, leaving the design of dynamic MMCs for a forthcoming publication.

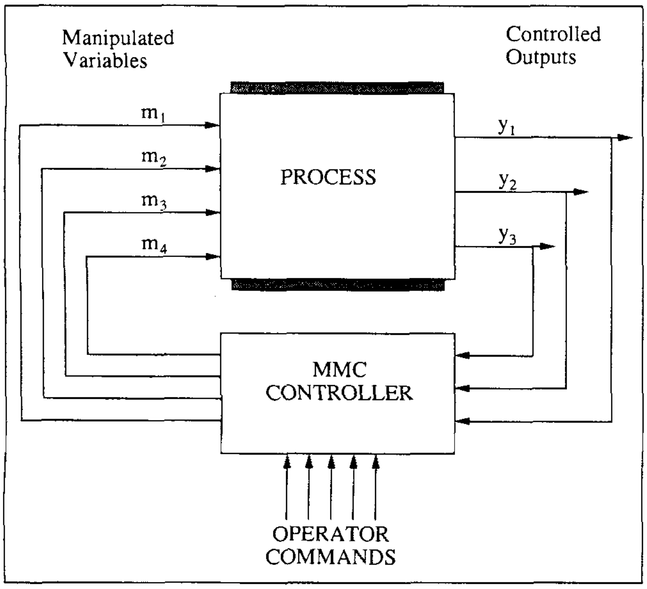

Figure 1. Feedback control loop of a multivariate process with four inputs and three outputs.

Structure and Functionality of Static MMC

As was mentioned earlier, the Modular Multivariable Controller (MMC) results from a design philosophy which is based on the direct solution of the controller design problem, using the lexicographic minimization of multiple objectives. In this section we will outline the structure and functionality of a static MMC and how these are determined by the lexicographic goal programming design approach.

Consider a process (Figure 1) with three outputs and four manipulated variables (inputs). We wish to keep the steady state behavior of the first two outputs, y_1 and y_2, at the corresponding desired set points, that is:

$$\text{Goal-1} \quad y_1 = y_{1,\text{sp}} \tag{14}$$

$$\text{Goal-2} \quad y_2 = y_{2,\text{sp}} \tag{15}$$

and keeping y_1 at its set point is more important than keeping y_2 at its set point. The third output should not drift outside a bounded region, that is:

$$\text{Goal-3} \quad y_3 \le y_{3,\text{up}} \tag{16a}$$

$$\text{Goal-4} \quad y_{3,\text{lo}} \le y_3 \tag{16b}$$

At all costs the controller must maintain y_2 above a minimum safe operating limit, that is:

$$\text{Goal-5} \quad y_{2,\text{lo}} \le y_2 \tag{17}$$

Goal-5 is a hard constraint (no deviation is permissible). The above statements indicate that the relative importance of Goals -1, -2, and -5 are as follows, with Goal-5 as the most important:

$$\text{Goal-5} > \text{Goal-1} > \text{Goal-2}$$

Engineering analysis also indicates that keeping y_3 below its upper bound is more important than maintaining it above its lower bound, and that both are of higher priority than either of the set point tracking objectives, Goal-1 and Goal-2. The relative priorities of Goals -3 and -4 are of little concern in this case as they are adjacent in ranking and could never be active (that is, at their equalities) at the same time. In fact, they could be combined into one goal with two nonconflicting limits. Leaving them separate merely offers greater flexibility for later re-prioritization. Thus, the complete lexicographic ranking of the five control objectives is:

$$\text{Goal-5} > \text{Goal-3} > \text{Goal-4} > \text{Goal-1} > \text{Goal-2}$$

In addition, we will assume that the manipulated variables cannot violate physical restrictions imposed by the capacity of the corresponding actuators (for example, valves), that is:

$$m_{i,\text{lo}} \le m_i \le m_{i,\text{up}} \qquad i = 1,2,3,4 \tag{18}$$

Finally, we will stipulate that the static open loop gains K_{ij} between output i and input j in the process model:

$$y_i = \sum_{j=1}^{4} K_{ij} \cdot m_j \qquad i = 1,2,3;\ j = 1,2,3,4 \tag{19}$$

may vary within a range of values (uncertainty in static gains).

$$K_{ij,\text{lo}} \le K_{ij} \le K_{ij,\text{up}} \qquad i = 1,2,3;\ j = 1,2,3,4 \tag{20}$$

Lexicographic goal programming formulation of the problem

The steady state values of the manipulated variables m_j, ($j = 1,2,3,4$) which satisfy the set of constraints, Eq. 14 through 19 for any values of K_{ij}'s in the range of Eq. 20, constitute the solution of the steady state control problem. This solution can be found through the solution of a series of LP problems, as will be described in this and the next section.

Convert the five control objectives into the following form:

$$g_1 \triangleq |y_1 - y_{1,\text{sp}}| \le b_1$$

$$g_2 \triangleq |y_2 - y_{2,\text{sp}}| \le b_2$$

$$g_3 \triangleq |\text{pos}(y_3 - y_{3,\text{up}})| \le b_3$$

$$g_4 \triangleq |\text{neg}(y_3 - y_{3,\text{lo}})| \le b_4$$

$$g_5 \triangleq |\text{neg}(y_2 - y_{2,\text{lo}})| \le b_5$$

where b_1, b_2, b_3, b_4, and b_5 are the aspiration levels of the five goals, and which for the static case can be all set to zero. Following the same development as earlier, but for the steady state control problem, we can derive the following lexicographic minimization:

Problem P3:

$$\underset{m}{\textbf{Min}}\ \boldsymbol{a}=(a_5,\ a_3,\ a_4,\ a_1,\ a_2) \tag{P3}$$

subject to:

$$g_1-\eta_1=0$$

$$g_2-\eta_2=0$$

$$g_3-\eta_3=0$$

$$g_4-\eta_4=0$$

$$g_5-\eta_5=0$$

$$m_{i,\text{lo}}\le m_i\le m_{i,\text{up}}\quad i=1\ 2,3,4$$

and y_i's and K_{ij}'s as defined in Eq. 19 and 20
$\rho_i=0$ because $b_i=0$, $i=1, 2, \ldots, 5$ making overachievement of objectives impossible (see the section on goal programming formulation of the controller design problem). The ordering of goals in the vector $\boldsymbol{a}$ in problem P3 signifies the relative priority of goals with a_5 corresponding to the most important goal, g_5.

Structure of the modular multivariable controller

Let us now see how the solution of the lexicographic minimization formulated in the section on lexicographic goal programming leads to the structure of the MMC.

Controller at Level I. Consider first the most important goal, g_5, and solve the following problem:

$$\underset{m_i, i=1,2,3,4}{\textbf{Min}}\quad a_5=\eta_5=|\text{neg}(y_2-y_{2,\text{lo}})|$$

$$=\left|\text{neg}\left(\sum_{j=1}^{4}K_{2j}\cdot m_j-y_{2,\text{lo}}\right)\right| \tag{S1}$$

$$\text{s.t. } m_{i,\text{lo}}\le m\le m_{i,\text{up}}\quad i=1,2,3,4$$

The statement of problem S-1 implies a controller with multiple inputs (that is, m_1, m_2, m_3, m_4) and a single control objective, that is, $y_2\ge y_{2,\text{lo}}$. Figure 2a shows this controller, which is denoted as CC-I. Instead of minimizing a_5 (or equivalently, achieving the control objective $y_2\ge y_{2,\text{lo}}$) by using all four degrees of freedom, we choose to assign a *primary manipulated variable*, say m_4, as the only degree of freedom in minimizing a_5. Consequently, problem S1 is converted into the following subproblem:

$$\underset{m_4}{\textbf{Min}}\ a_5=\eta_5=|\text{neg}(y_2-y_{2,\text{lo}})|$$

$$=\left|\text{neg}(K_{24}\cdot m_4+\sum_{j=1}^{3}K_{2j}\cdot m_j^{\text{II}}-y_{2,\text{lo}})\right| \tag{S1a}$$

$$\text{s.t. } m_{4,\text{lo}}\le m_4\le m_{4,\text{up}}$$

The manipulations m_1, m_2, and m_3, have been given specific values, m_1^{II}, m_2^{II}, and m_3^{II} called Manipulated-Variable (MV) targets. The superscript II denotes that these values are coming from the coordinated controller at Level-II, which corresponds to the next most important goal, as we will see in the following paragraphs. In subsequent sections we will also describe how the primary manipulated variables are selected and how the MV-targets are set. For the time being it suffices to point out that the primary manipulated variable is selected using a concise procedure (see Section on selection of primary manipulated variables) which accounts for the modeling uncertainty in the open-loop static gains and their impact on the performance of the static controller, attempting to achieve a specific goal.

It is easy to see that the solution to Subproblem S1a is given by the value(s) of m_4 in the intersection set of the following constraints:

$$m_4\ge m_4^*\ (\text{if } K_{24}>0)\text{ or } m_4\le m_4^*\ \ (\text{if } K_{24}<0) \tag{21a}$$

and

$$m_{4,lo}\le m_4\le m_{4,\text{up}} \tag{21b}$$

where,

$$m_4^*=\frac{1}{K_{24}}\cdot\left(y_{2,\text{lo}}-\sum_{j=1}^{3}K_{2j}\cdot m_j^{\text{II}}\right) \tag{22}$$

Furthermore, note that variable m_4, like the other three manipulations, possesses an MV-target, m_4^{II}, which is also provided to the controller CC-I (see Figure 2a). Let us examine more closely the characteristics of the solution to problem S1a, as given by the intersection of constraints of Eqs. 21a and 21b:

Case 1: Feasible Solution(s). The intersection set of constraints of Eqs. 21a and 21b is not null, and all the values of m_4 in this set are acceptable solutions. Controller CC-I produces new MV-targets for the manipulated variables, as follows:

- If m_4^{II} satisfies Eqs. 21a and 21b then m_4^{II} is a trivial solution to problem S1a. The controller CC-I sets $m_4^{\text{I}}=m_4^{\text{II}}$ and $m_i^{\text{I}}=m_i^{\text{II}}$, $i=1,2,3$ and passes the new MV-targets to the actuators for implementation.
- If m_4^{II} does not satisfy Eq. 21a, then controller CC-I sets $m_4^{\text{I}}=m_4^*$ and $m_i^{\text{I}}=m_i^{\text{II}}$, $i=1,2,3$ and passes the new MV-targets to the actuators.

Case 2: Infeasible Solution. The null intersection of constraints Eqs. 21a and 21b implies that m_4^* is outside the allowable range $[m_{4,\text{lo}},\ m_{4,\text{up}}]$. Thus, the primary manipulation m_4 will saturate at $m_{4,\text{lo}}$ (or $m_{4,\text{up}}$) without satisfying $y_2\ge y_{2,\text{lo}}$. Controller CC-I sets $m_4^{\text{I}}=m_{4,\text{lo}}$ (or $m_{4,\text{up}}$) depending on the sign of K_{24} and introduces a *second primary manipulated variable* (a new degree of freedom), say m_2, by formulating the following problem:

$$\underset{m_2}{\textbf{Min}}\ a_5=\eta_5=|\text{neg}(y_2-y_{2,\text{lo}})|$$

$$=\left|\text{neg}(K_{22}\cdot m_2+K_{24}\cdot m_4^{\text{I}}+\sum_{j=1,3}K_{2j}\cdot m_j^{\text{II}}-y_{2,\text{lo}})\right| \tag{S-1b}$$

$$\text{s.t. } m_{2,\text{lo}}\le m_2\le m_{2,\text{up}}$$

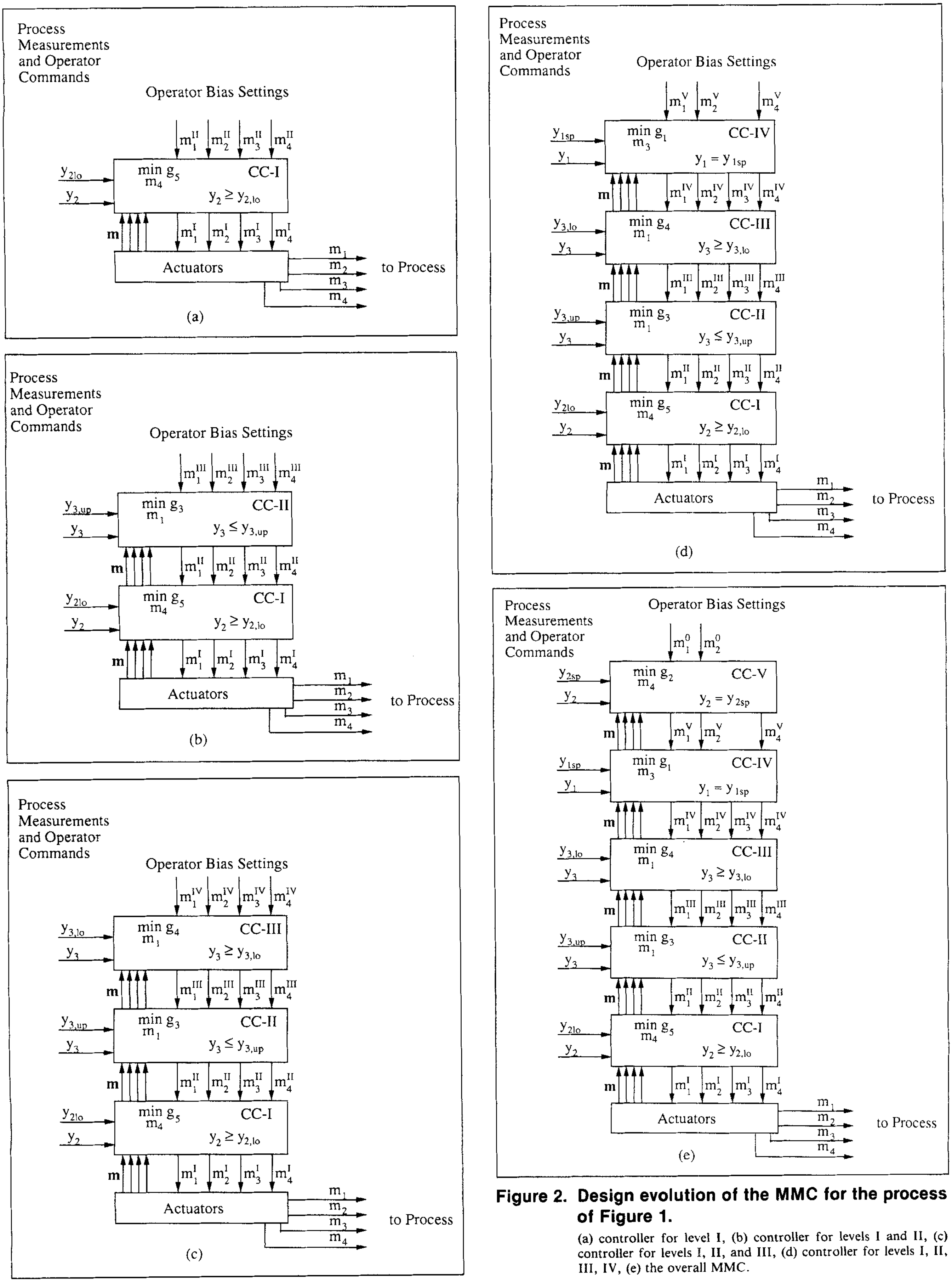

Figure 2. Design evolution of the MMC for the process of Figure 1.

(a) controller for level I, (b) controller for levels I and II, (c) controller for levels I, II, and III, (d) controller for levels I, II, III, IV, (e) the overall MMC.

A similar analysis, as above with m_4, may produce one of the following situations:

- $m_2^{\mathrm{I}} = \frac{1}{K_{22}} \cdot \left(y_{2,\mathrm{lo}} \cdot K_{24} \cdot m_4^{\mathrm{I}} - \sum_{j=1,3} K_{2j} \cdot m_j^{\mathrm{II}} \right)$ or

- $m_2^{\mathrm{I}} = m_{2,\mathrm{lo}}$ or $m_{2,\mathrm{up}}$, without satisfying $y_2 \geq y_{2,\mathrm{lo}}$, that is, Min $a_5 \neq 0$. In this case a third primary manipulated variable is needed to satisfy the objective.

Depending on the situation, controller CC-I sets m_2^{I} to the appropriate value, and along with m_4^{I} (from problem S1a) and $m_i^{\mathrm{I}} = m_i^{\mathrm{II}}$, $i = 1,3$, sends the new MV-targets to the actuators for implementation. For the purposes of this example we will assume that m_4 is sufficient to satisfy $y_2 \geq y_{2,\mathrm{lo}}$. Controller CC-I is identical in character to the *Coordinated Controller*, proposed by Brosilow (Popiel et al., 1986; Brosilow, 1990).

Controller at Level-II. In attempting to achieve the next most important goal, g_3, the definition of the lexicographic minimum requires that the more important goal, g_5, remains satisfied. Thus we pose the following problem:

$$\underset{m_i, i=1,2,3,4}{\mathbf{Min}} \quad a_3 = \eta_3 = |\mathrm{pos}(y_3 - y_{3,\mathrm{up}})|$$

$$= \left| \mathrm{pos}\left(\sum_{j=1}^{4} K_{3j} \cdot m_j - y_{3,\mathrm{up}} \right) \right| \quad \text{(S2)}$$

$$\text{s.t. } a_5 = 0$$

$$m_{i,\mathrm{lo}} \leq m \leq m_{i,\mathrm{up}} \quad i = 1,2,3,4$$

Problem S2 suggests a controller with multiple inputs and a single control objective, that is, $y_3 \leq y_{3,\mathrm{up}}$. This controller, denoted as CC-II (see Figure 2b), interacts with CC-I which maintains the satisfaction of the more important goal, $a_5 = 0$, leading to an overall multivariable controller with two control objectives. Let m_1 be the primary manipulated variable for CC-II. Then, problem S2 can take one of the following forms, depending on the solution reached by problem S1a:

Case 1. m_4^I is Unsaturated. This case corresponds to the cases where problem S1a possesses a feasible solution with m_4 as the only primary manipulated variable. In such cases, m_4 is bounded in the range $[m_4^*, m_{4,\mathrm{up}}]$ or $[m_{4,\mathrm{lo}}, m_4^*]$, depending on the sign of K_{24}, and the problem S2 takes on the following form:

$$\underset{m_1, m_4}{\mathbf{Min}} \; a_3 = \eta_3 = |\mathrm{pos}(y_3 - y_{3,\mathrm{up}})|$$

$$= \left| \mathrm{pos}(K_{31} \cdot m_1 + K_{34} \cdot m_4 + \sum_{j=2,3} K_{3j} \cdot m_j^{\mathrm{III}} - y_{3,\mathrm{up}}) \right| \quad \text{(S2a)}$$

$$\text{s.t. } \max[m_{4,\mathrm{lo}}, m_4^*] \leq m_4 \leq m_{4,\mathrm{up}} \; (\text{or } m_{4,\mathrm{lo}} \leq m_4 \leq \max[m_{4,\mathrm{up}} \cdot m_4^*])$$

$$m_{1,\mathrm{lo}} \leq m_1 \leq m_{1,\mathrm{up}}$$

Case 2. m_4^I is Saturated. This case corresponds to the case where no feasible solution exists for problem S1a, and CC-I requires a second primary manipulated variable, say m_2, to achieve Min $a_5 = 0$. In such cases, problem S2 takes on the following form:

$$\underset{m_1, m_2}{\mathbf{Min}} \; a_3 = \eta_3 = |\mathrm{pos}(y_3 - y_{3,\mathrm{up}})|$$

$$= |\mathrm{pos}(K_{31} \cdot m_1 + K_{32} \cdot m_2 + K_{34} \cdot m_4^{\mathrm{I}} + K_{33} \cdot m_3^{\mathrm{III}} - y_{3,\mathrm{up}})| \quad \text{(S2b)}$$

$$\text{s.t. } \max[m_{2,\mathrm{lo}}, m_2^*] \leq m_2 \leq m_{2,\mathrm{up}} \; (\text{or } m_{2,\mathrm{lo}} \leq m_2 \leq \max[m_{2,\mathrm{up}}, m_2^*])$$

$$m_{1,\mathrm{lo}} \leq m_1 \leq m_{1,\mathrm{up}}$$

(Note: m_2^* in Eq. S2b is computed from the constraint $y_3 \leq y_{3,\mathrm{up}}$ in problem S1b in a manner analogous to that of m_4^* in problem S1a.) Problems S2a and S2b indicate very clearly the interaction between the coordinated controllers CC-I and CC-II, which can be described as follows:

- Controller CC-I determines the range of allowable MV-target values for its own primary manipulated variable (for example, m_4, or m_4 and m_2), so that its corresponding control objective, g_5, is satisfied. Then it passes these allowable values to CC-II.
- Controller CC-II determines the range of allowable MV-target values for its own primary manipulation(s), in such a way that its corresponding objective, g_3, is satisfied and that the conditions passed on by CC-I are not violated.

This coordinated action of CC-I and CC-II produces the MV-targets, m_i^{I}, $i = 1,2,3,4$, which are sent to the actuators for implementation.

Controller at Level-III. In a similar manner we can formulate the minimization problem corresponding to controller CC-III for goal g_4 (see Figure 2c). The manipulated variable m_1 can also be assigned as primary to CC-III since goals g_3 and g_4 can never be in conflict. Alternately, CC-II and CC-III could take on different primary manipulated variables. This distinction could be advantageous especially if $y_{3,\mathrm{lo}}$ and $y_{3,\mathrm{up}}$ represent widely differing conditions with significantly different process dynamics. The minimization problem for CC-III can take on one of the following two forms (depending on the solution of problem S1a).

Case 1 m_4^I is Unsaturated.

$$\underset{m_1, m_4}{\mathbf{Min}} \; a_4 = \eta_4 = |\mathrm{neg}(y_3 - y_{3,\mathrm{lo}})|$$

$$= \left| \mathrm{neg}\left(K_{31} \cdot m_1 + K_{34} \cdot m_4 + \sum_{j=2,3} K_{3j} \cdot m_j^{\mathrm{IV}} - y_{3,\mathrm{lo}} \right) \right| \quad \text{(S3a)}$$

$$\text{s.t. } \max[m_{4,\mathrm{lo}}, m_4^*] \leq m_4 \leq m_{4,\mathrm{up}} \; (\text{or } m_{4,\mathrm{lo}} \leq m_4 \leq \max[m_{4,\mathrm{up}}, m_4^*])$$

$$m_{1,\mathrm{lo}} \leq m_1 \leq m_{1,\mathrm{up}}$$

Case 1 m_4^I is Saturated.

$$\underset{m_1, m_2}{\mathbf{Min}} \; a_4 = \eta_4 = |\mathrm{neg}(y_3 - y_{3,\mathrm{lo}})|$$

$$= |\mathrm{neg}(K_{31} \cdot m_1 + K_{32} \cdot m_2 + K_{34} \cdot m_4^{\mathrm{I}} + K_{33} \cdot m_3^{\mathrm{IV}} - y_{3,\mathrm{lo}})| \quad \text{(S3b)}$$

s.t. $\max[m_{2,lo}, m_2^*] \leq m_2 \leq m_{2,up}$ (or $m_{2,lo} \leq m_2 \leq \max[m_{2,up}, m_2^*]$)

$$m_{1,lo} \leq m_1 \leq m_{1,up}$$

Controller at Level-IV. Goal g_1 is an equality, or set point goal. With m_3 the primary manipulated variable, the corresponding minimization problem (assuming that m_4 and m_1 do not saturate) is given by:

$$\underset{m_1, m_3, m_4}{\textbf{Min}} \; a_1 = \eta_1 = |y_1 - y_{1,sp}|$$

$$= \left| \sum_{j=1,3,4} K_{1j} \cdot m_j + K_{12} \cdot m_2^V - y_{1,sp} \right| \quad \text{(S4a)}$$

s.t. $\max[m_{4,lo}, m_4^*] \leq m_4 \leq m_{4,up}$ (or $m_{4,lo} \leq m_4 \leq \max[m_{4,up}, m_4^*]$)

$\max[m_{1,lo}, m_1^*] \leq m_1 \leq m_{1,up}$ (or $m_{1,lo} \leq m_1 \leq \max[m_{1,up}, m_1^*]$)

$$m_{3,lo} \leq m_3 \leq m_{3,up}$$

If a feasible solution exists for Problem S4a, coordinated controller CC-IV (see Figure 2d) sets:

$$m_3^{IV} = \frac{1}{K_{13}} \cdot \left(y_{1,sp} - \sum_{j=1,2,4} K_{1j} \cdot m_j^{IV} \right) \quad (23)$$

$m_1^{IV} = m_1^V$ and $m_4^{IV} = m_4^V$ (if the constraints in Problem S4a are trivially satisfied)

$$m_2^{IV} = m_2^V$$

The MV-target values, m_1^{IV} and m_4^{IV}, will be different from those of m_1^V and m_4^V, respectively, if the corresponding constraints in Eq. S-4a are not trivially satisfied, for example, $m_4^{IV} = m_4^*$, or $m_{4,up}$, or $m_{4,lo}$, as the case may be.

Controller at Level-V. Goal-2 is also an equality constraint. If m_4 is the selected primary manipulated variable, then the coordinated controller CC-V (Figure 2e) corresponds to the following minimization problem:

$$\underset{m_1, m_3, m_4}{\textbf{Min}} \; a_2 = \eta_2 = |y_2 - y_{2,sp}|$$

$$= \left| \sum_{j=1,2,4} K_{2j} \cdot m_j + K_{23} \cdot m_3^{IV} - y_{2,sp} \right| \quad \text{(S5a)}$$

s.t. $\max[m_{4,lo}, m_4^*] \leq m_4 \leq m_{4,up}$ (or $m_{4,lo} \leq m_4 \leq \max[m_{4,up}, m_4^*]$)

$\max[m_{1,lo}, m_1^*] \leq m_1 \leq m_{1,up}$ (or $m_{1,lo} \leq m_1 \leq \max[m_{1,up}, m_1^*]$)

where m_3^{IV} is parametrically expressed by Eq. 23. The form of Problem S5a suggests that coordinated controller CC-V will fix the MV-target value of m_4. Given that the MV-target of m_3 is fixed by CC-IV (Eq. 23), then the actions of CC-V are parameterized in terms of the MV-targets given for m_1 and m_2 (see Figure 2e). Therefore, controller CC-V produces MV-target values, which are given by:

$$m_1^V = m_1^0, \quad m_2^V = m_2^0, \quad \text{and}$$

$$m_4^V = \frac{1}{K_{24}} \cdot \left(y_{2,sp} - \sum_{j=1}^{3} K_{2j} \cdot m_j^V \right) \quad (24)$$

The complete arrangement of the five coordinated controllers, shown in Figure 2e, represents the MMC, which solves the set of equality and inequality constraints given by Eqs. 14 through 19. We will discuss later how the MMC handles uncertainty, Eq. 20, by appropriately selecting the primary manipulated variables.

Examples of numerical calculations are presented in Appendix B.

Some notes on the operational characteristics of the procedure generating the MMC

The procedure described in the previous section generates the structure of the MMC and defines its operational character. It should be pointed out that the main thrust of the lexicographic minimization, through a series of LP problems, is to solve a set of simultaneous linear equalities and inequalities (defining the control goals); a task known to be resolved (Ignizio, 1983) through LP formulations. These LP problems using the simplex method, with the added stipulation that η_i and ρ_i *cannot become basic variables simultaneously*, since over- and underachievement of a goal are not simultaneously possible. Also, the solution of an LP problem can always start from a feasible point, which is the solution of the LP problem at the previous stage (that is, priority level). Thus, Subproblem S-2a starts from the solution of Subproblem S-1a, which has yielded $a_5 = 0$, that is, a feasible point. In this case, the feasible point is on a feasible line segment because the subproblem was an inequality defining a region, not a point. Let us now examine the implications of the solution procedure, described in the previous section, in terms of its impact on defining an acceptable multivariable controller.

Situations of Under-Specified Control. The controller design problem for the process in Figure 1 is generally under-specified. There are two (or three, if y_3 is at a bound) independent objectives (note that satisfaction of Goal-2 entails satisfaction of Goal-5) and four manipulated variables (assuming no saturation or failure of manipulations). If m_3 and m_4 are the selected primary manipulated variables for the equality goals g_1 and g_2 respectively, the values of the remaining manipulations m_1 and m_2 are provided as *MV-targets* to the MMC structure through the least important coordinated controller (controller CC-V in Figure 2e). These MV-targets can be set manually by the operators or automatically by an external supervisory optimizer. In the absence of saturated manipulated variables the MV-targets are passed unchanged down the hierarchy of coordinated controllers, and are transferred to the actuators. If, on the other hand, primary m_4 in coordinated controller CC-V saturates in its effort to satisfy goal g_2, CC-V will employ its second primary manipulated variable in addition to m_4, say m_2. Both m_4 and m_2 are needed to control y_2, so the externally supplied MV-target value on m_2 is ignored.

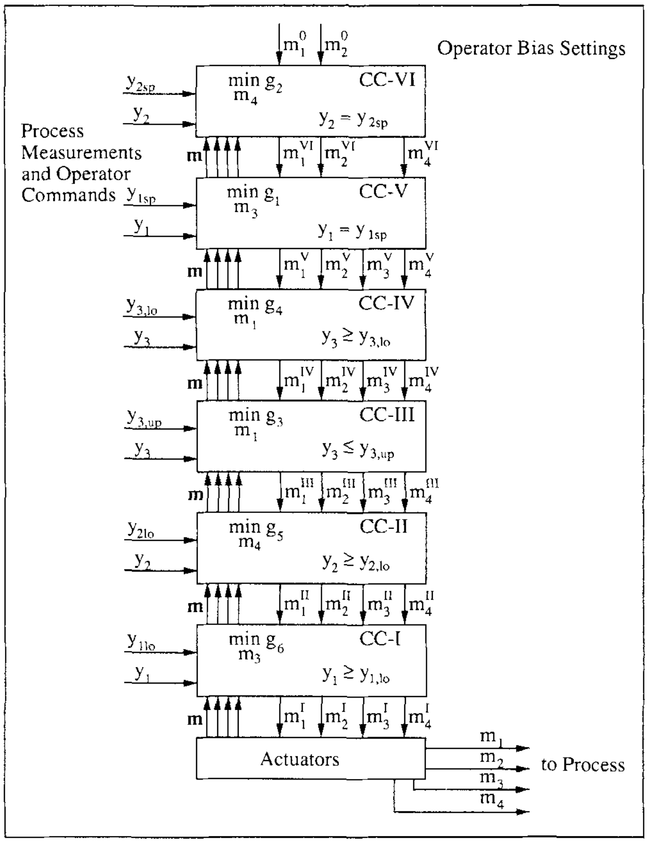

Figure 3. MMC for the process of Figure 1 with additional constraint on y_1.

It should be noted that the desired value for m_2 is not sent to the actuator by CC-V, but instead is passed on to the more important coordinated controllers (for example, CC-IV, CC-III, CC-II and CC-I) as a *MV-target request*. This is determined by the character of the solution procedure for finding the lexicographic minimum, that is, no objective of higher priority should be "sacrificed" in favor of an objective of lesser importance. Thus, if the *MV-target request* on m_2 sent by CC-V does not adversely affect the more important goals, g_1, g_4, g_3 and g_5, then it will pass on to the actuator. It is this feature of MMC that renders this controller as a true multivariable one that accounts for all interactions.

Situations of Exactly Specified Control. In attempting to satisfy equality goals g_1 and g_2, it is possible that both the primaries m_3 and m_4 saturate. If this is the case, the MV-targets on both degrees of freedom, that is, m_1 and m_2, are ignored and the control system becomes exactly specified. The MV-target requests on m_1 and m_2 are forwarded by coordinated controllers CC-V and CC-IV down the hierarchy through CC-III, CC-II and CC-I, and if none of the inequality goals g_4, g_3, and g_5 is violated they are implemented on the actuators.

Situations of Overspecified Control. As manipulated variables saturate, a situation of over-specified control may arise. Consider the case where primary manipulated variables m_4 and m_3 in CC-V and CC-IV, respectively, have saturated. Then, we have no degrees of freedom left, since m_1 and m_4 are set by the equality goals g_1 and g_2. If in addition y_3 is at one of its bounds, for example, $y_{3,lo}$, then we have an overspecified control situation (three equations, two variables). In such cases, the MMC operates as follows:

- The MV-target requests on m_1 and m_2 are passed on to CC-III which is assigned to goal g_4, that is, $y_3 \le y_{3,lo}$. Since $y_3 = y_{3,lo}$ the MV-target requests (satisfying equalities g_1 and g_2) cannot be "honored" by CC-III since they would violate the goal g_4.
- The MMC will relax the least important goal, g_2, and will free a degree of freedom, thus enabling the satisfaction of goal g_4. In other words, CC-III will generate new MV-target requests, which will be sent to CC-II and from there to CC-I.
- If the relaxation of goal g_2, that is, $y_2 = y_{2,sp}$, leads to the violation of goal g_5, the inequality $y_2 \ge y_{2,lo}$, then the next least important objective, that is, g_1, will be relaxed and the degree of freedom will be assigned to CC-I.
- The relaxation of goal g_1, that is $y_1 = y_{1,sp}$ may allow y_1 to "wander around" without direct control. If that is unacceptable, then it is clear that the original statement of the control problem was incomplete, since it lacked a sixth goal constraining the behavior of y_1. The modular structure of MMC allows the easy addition of missing objectives. If $y_1 \le y_{1,up}$ is a satisfactory bounding of y_1, then it generates a sixth goal which in relative priority we assume is before goal g_5. The resulting MMC is shown in Figure 3.

It is clear from the above description that operationally the MMC always identifies a square control system corresponding to the most important objectives, and does not require ad hoc rules to handle saturations (or failures) of manipulated variables.

Model Uncertainty and its Impact on the Design of Robust Static MMCs

All feedback controllers are designed on the basis of an assumed process model (Morari and Zafiriou, 1989). The study of the impact that uncertainty in the assumed model could have on the performance of a feedback controller has been one of the most important developments in control theory since the mid-1970's. Consequently, the design methodology leading to an MMC should account for model uncertainty and should guarantee the robustness of the resulting controller. In this section we will see how the robustness issues of a static MMC are taken into account and how they are used to determine the primary manipulated variables corresponding to specific goals.

The robustness of the static MMC is its insensitivity to the errors of the static gains, as these have been assumed to vary within certain ranges, for example, such as those given by Eq. 20. The robustness of a multivariable control system can be generally expressed by some type of condition number for the matrix of open-loop static gains with structured or unstructured, norm-bounded uncertainty. These condition numbers, which signify the "size" of gain variations that could lead to system singularities, can be interpreted within an abstract mathematical framework, but cannot in general be directly related to or explained in physical/engineering terms. For instance, the disturbance condition number (Skogestad and Morari, 1987b) describes how well the direction of a disturbance is aligned with the direction of the best response for the plant, or the plant and compensator. Low numbers are good and high numbers are bad but from an engineering standpoint the disturbance condition number is an abstract guideline except in the special case where the plant is well-aligned with the likely disturbances (for example, Skogestad and Morari, 1987a,b). The disturbance condition number is bounded by the plant condition number, which indicates the degree of "directedness" of the plant. Plants with strong directional characteristics are often hard to control. Thus model uncertainty has always been handled implicitly by past controller design approaches. The robustness of the controller has been associated with abstract quantities.

A measure of closed-loop robustness

Consider the steady-state "closed-loop" gain, G, between a given primary manipulated variable (input) and a given controlled output provided that all more important controlled outputs have remained constant under feedback control (see two sections ahead for a formal definition of G). Let $|G|_{max}$ and $|G|_{min}$ be the maximum and minimum magnitudes that the closed-loop gain can take on, as the open-loop static gains of a multivariable system vary within the respective ranges of their uncertainties. Then, the ratio C_g defined by

$$C_g = \frac{|G|_{max}}{|G|_{min}} \tag{26}$$

constitutes an explicit measure of the closed-loop robustness between the given input and output. Note that if the static closed-loop gain can change sign, then $|G|_{min}=0$, and the value of C_g becomes infinity. The *robustness measure* C_g is a function of both the open-loop nominal gain matrix and the matrix of possible errors on the gain matrix. A value of $C_g=1$ indicates best robustness, because the possible errors, if any, of the open-loop gain matrix have no effect on the closed-loop gain. Likewise a value of $C_g=\infty$ indicates that the true value for G may be of a different sign to the modelled (nominal) value, a situation of minimum robustness.

The *robustness measure* C_g can emulate the same qualitative results as any condition number based on the additive uncertainty of static gain matrices, but its interpretation in engineering terms is much more direct. The difficulty with using C_g comes from its computation. If every static open-loop gain in an $n \times n$ multivariable system (that is, the system governing the nth most important primary) can vary between an upper and a lower bound, the computation of $|G|_{max}$ and $|G|_{min}$ could potentially involve the evaluation of 2^{n^2} possible solutions (see Appendix A); clearly an intractable numerical task even for low-order systems. In subsequent sections we will discuss an approach based on the simplex algorithm, which although for pathological cases could be intractable, has been proven to be much more efficient in practice than its pathological complexity suggests.

Robustness of a static MMC design

As was outlined in the section discussing the structure of the MMC, each goal of the MMC is achieved through the manipulation of the corresponding *primary manipulated variables*. Consequently, the closed-loop robustness of the corresponding coordinated controller can be measured by C_g between the specific goal and the primary manipulated variable (primary). For example, consider the Subproblem S-1a. To minimize the impact of model uncertainty on the closed-loop achievement of the most important objective, g_5, we could use as primary the one with the smallest C_g. This selection implies the computation of four C_g's for 1×1 systems, clearly a trivial numerical task. For Subproblem S-2a, the selection of the primary manipulated variable for Goal-3 with the minimum C_g involves the evaluation of three C_g's for 2×2 systems (since Goal-5 must be kept satisfied). Thus, in general the selection of the nth primary manipulated variable involves the computation of C_g for an $n \times n$ system, which could be numerically intractable. Also, the numerical value of the robustness measure C_g increases as we move to less and less important goals, as a result of the multiplicative interactions between the open-loop gain uncertainties in a multivariable system. Several useful consequences arise from this sequential examination of an MMCs robustness to modelling errors:

- The impact of gain uncertainty on a more important control goal is smaller than for a less important goal. Thus, the coordinated controller handling the achievement of an important control goal is expected to be more robust than the coordinated controller handling a goal of lesser importance.
- As we move to less important goals, the computational load for evaluating the corresponding C_g's increases. Any approximations that might be introduced to facilitate the computations, resulting in more conservative measures, are less critical because they impact on the quality of control for goals of lesser importance.

Computing the robustness measure C_g for static MMC

Consider a plant of n controlled outputs $y_1, y_2, \ldots, y_n$, sorted in order of decreasing importance, and p manipulated variables $m_1, m_2, \ldots, m_p$. Let $\boldsymbol{K}=\boldsymbol{C}\cdot\boldsymbol{\Gamma}$ (where $\boldsymbol{C}$ and $\boldsymbol{\Gamma}$ are defined in Eq. 5) represent the matrix of the open-loop gains for the plant, that is,

$$\boldsymbol{K} = [K_{ij}] = \left(\frac{\delta y_i}{\delta m_j}\right)_{m_q q=1,\ldots,p\ q \neq j} \tag{27}$$

If we designate the columns of $\boldsymbol{K}$ by $\boldsymbol{k}_i$, $(i=1, 2, \ldots p)$, that is

$$\boldsymbol{K} = [\boldsymbol{k}_1 \boldsymbol{k}_2 \ldots \boldsymbol{k}_p] \tag{28}$$

then we can define $\boldsymbol{K}^i_{p_1 p_2 \ldots p_i}$ to be the $i \times i$ matrix made up of the first i rows of the matrix $[\boldsymbol{k}_{p_1}\boldsymbol{k}_{p_2}\ldots\boldsymbol{k}_{p_i}]_{nxi}$, where the subscript p_m denotes the primary manipulated variable for the mth controlled variable.

Define the "closed-loop" gain, G_i, between a controlled output y_i and its primary manipulated variable m_{p_i}, by:

$$G_i = \left[\frac{\delta y_i}{\delta m_{p_i}}\right] \tag{29}$$

assuming that all more important outputs are held constant, that is, y_k constant, $k=1, 2, \ldots, i-1$ and that all manipulated variables that are not already designated as primaries are held constant, that is, m_h constant, $h \neq p_1, \ldots, p_{i-1}$. It is easy to show (see Appendix A) that:

$$G_i = \frac{\mathrm{Det}(K^i_{p_1p_2\ldots p_i})}{\mathrm{Det}(K^{i-1}_{p_1p_2\ldots p_{i-1}})} \tag{30}$$

where $K^i_{p_1p_2\ldots p_i}$ is defined above.

From Eq. 30 we see that the closed-loop gain between the ith controlled output, y_i, and its corresponding primary manipulated variable, m_{p_i}, is given as the ratio of two determinants. Let us define:

$$A = K^i_{p_1p_2\ldots p_i} \quad \text{and} \quad A' = K^{i-1}_{p_1p_2\ldots p_{i-1}} \tag{31}$$

$$\text{so that} \quad G_i = \frac{\det(A)}{\det(A')} \tag{32}$$

It is clear that A' is the largest principal minor of A.

Assume that the elements of the matrix K, K_{ij} can vary (either in a correlated or an uncorrelated fashion) between an upper and a lower bound, reflecting the uncertainty in their modelled values. A range of values for G_i, $\det(A)$ and $\det(A')$ will be possible. Consider two cases:

- If the range of values of $\det(A)$ crosses zero, that is, if $\det(A)$ can change sign under the variation of the elements of K, then from Eq. 32, $|G_i|_{\min} = 0$ and $C_g(G_i) = \infty$.
- If $\det(A')$ can change sign under the variation of the elements of K, then from Eq. 32, $|G_i|_{\max} = \infty$ and $C_g(G_i) = \infty$.

Depending on the assumed structure of gain perturbations, we can distinguish the following three cases for the estimation of the robustness measure, C_g.

Case 1. Uncorrelated Perturbations of Gains (Unstructured Uncertainty). Each K_{ij} is assumed to vary independently within a range of values (see inequalities Eq. 20). In this case one can show that the following two theorems hold (see Appendix A for proofs).

Theorem 1. If each of the i^2 elements of A (where A is defined above in terms of K) is allowed to vary independently between a maximum and a minimum value, then the extreme values of the determinant of A, that is, max(det(A)) and min(det(A)), will occur at extreme values (maximum or minimum) of the elements of A.

Theorem 2. If each of the i^2 elements of A (where both A and A' are defined above in terms of K) is allowed to vary independently between a maximum and a minimum value, and both $\det(A)$ and $\det(A')$ cannot change sign under that variation, then the extreme closed-loop gains between the ith controlled output, y_i, and its corresponding primary manipulated variable, m_{p_i}, that is, $|G_i|_{\max}$ and $|G_i|_{\min}$ will occur at extreme values (maximum and minimum) of the elements of A.

Theorems 1 and 2 allow us to find the solution using a linear program (LP), as outlined in the Appendix A. While we are guaranteed to find a correct solution, this is a computationally NP-complete problem. Therefore, there exists no computationally tractable algorithm to evaluate $C_g(G_i)$ for the pathological case. However, in practice, the algorithm described in Appendix A, like all LP solvers (for example, the simplex method), proves to be much more efficient than the pathological complexity suggests.

The computation of $C_g(G_i)$ becomes potentially "expensive" for the less important control goals of a multivariable system. Therefore, approximate or bounded estimates of $C_g(G_i)$ for the less important objectives of large-size multivariable systems are perfectly acceptable and do not result in overly conservative designs. Specifically, if an estimate of $C_g(G_i)$ is required or if the computation of $C_g(G_i)$ is numerically intractable the following theorem (for the proof see Appendix A) provides lower and upper bounds on $C_g(G_i)$.

Theorem 3. The Robustness measure, C_g, defined by Eq. 21 satisfies the following conditions:

$$C_g(G_i) \le \frac{\max[\det(A)] - \max[\det(A')]}{\min[\det(A)] \cdot \min[\det(A')]} \tag{33a}$$

$$C_g(G_i) \ge \frac{Q_{\max}}{Q_{min}} \tag{33b}$$

where Q is the set

$$\left\{ \frac{\max[\det(A)]}{\max[\det(A')]}, \frac{\min[\det(A)]}{\min[\det(A')]}, G_{i,\text{nominal}} \right\}$$

It should be noted that the computation of the extreme values of $\det(A)$ is a simpler problem than the problem of finding the extreme values of G_i because of the linear dependence of $\det(A)$ on the elements of A shown in the proof of Theorem 1. Thus the bounds, Eqs. 33a and 33b on C_g can be easily computed.

Case 2. Unidimensional Correlation of Gain Perturbations (Structured Uncertainty). The completely uncorrelated perturbation of K_{ij}'s, assumed in Case 1 above, is a rather unrealistic scenario. More likely, the perturbation of a static gain is correlated with those of other gains along the same row or column. A typical example is the gain uncertainty structure of the Shell heavy fractionator, where the gains along the same column (effect of one manipulated input) vary in a correlated manner. In such instances the results given above by Theorems 1, 2, and 3 also apply (see Appendix A). It should be noted that in these cases of correlated gain perturbations the complexity of evaluating the robustness measure $C_g(G_i)$ is reduced. If m gains on the same row or column are correlated, there is only one independent variable instead of m. The computation of $C_g(G_i)$ is of complexity 2^{i^2+1-m} instead of 2^{i^2}.

Case 3. Multi-Dimensional Correlation of Gain Perturbations. In a few instances, the perturbations of a static gain may be correlated with those of other gains distributed over several rows and columns. The resulting robustness measure becomes, generally, a nonlinear function of the correlated perturbations and the results of Theorems 1, 2 and 3 do not apply. In these cases it can be easily shown that the extrema $|G_i|_{\max}$ and $|G_i|_{\min}$ are located at two of the solutions of the mth order polynomial:

$$P_m(\epsilon) = \frac{\delta G_i}{\delta \epsilon} = 0$$

where m is the number of correlated gains and ϵ is the common perturbation parameter of the correlated gains. A nonlinear optimization algorithm would be required to compute the robustness measure.

Selection of primary manipulated variables

Coordinated controllers in an MMC (or equivalently, control objectives) are assigned their corresponding primary manipulated variables sequentially in order of decreasing relative importance. The selection of primary manipulated variables converts a nonsquare system with i objectives (the current controlled output and all those that have higher priority) and p manipulated variables ($i-1$ already selected as primaries), to a square $i \times i$ system, which has a unique solution to any steady state set of constraint specifications.

The design of a static multivariable controller is primarily driven by the system's ability to maintain a broad operating range without saturating the manipulated variables. The primary manipulated variables of an MMC will saturate for one of two reasons. First, a primary will saturate when the operating objectives are unrealizable using only the primary manipulated variables. This situation is minimized if the primaries are selected according to the following criterion:

Criterion 1. Maximize the operating range where the primary in question is free, that is, not saturated. A direct consequence of Criterion 1 is the selection of a primary with the largest closed-loop gain. Second, a primary will saturate in the wrong direction if the true closed-loop gain is of a different sign than the modeled closed-loop gain. This situation is minimized if the primaries are selected according to the criterion:

Criterion 2. Minimize the effect of steady state gain errors. The direct consequence of Criterion 2 is to select the primary in such a way that the corresponding measure of robustness, C_g, has the minimum value.

The selections of primary manipulated variables suggested by Criteria 1 and 2 are independent of each other, and in general not commensurable. In order to achieve the best set of primary manipulated variables for a MMC structure, we do not resolve conflicts at the level of individual coordinated controllers, but we maintain a tree of options. Thus:

- If criteria 1 and 2 suggest different primaries at priority level i, both options are retained and are evaluated at the next priority level $i+1$.
- If criterion 1 or 2 produces two or more equally good primary options, all equal options are retained for the next priority level.
- If several options of primaries are available after the assignment of primaries at the lowest priority level, the selection of the preferred set of primaries is based on the engineer's subjective judgement as to whether higher gains are preferable to broad robustness or vice versa.

By allowing branching in the primaries selection procedure, the resulting structure is lexicographically optimal and equivalent to any other nonsequential procedure. But, the sequential procedure used during the design of an MMC reveals important "local" information which significantly improves the human designer's subjective judgement in cases of multiple alternatives.

The closed-loop gain of each primary depends on the primary selections that were made for the more important objectives before it. In the event of an actuator saturating, sensor or actuator failure (see the section on tolerance to sensor and actuator features), or an inequality objective becoming active, the loss of a given primary changes the closed-loop gains of the primaries for all of those objectives that are less important. To allow for these losses of a degree of freedom, not only the first choice but all of the possible choices for a given primary must be considered when choosing the primaries for the less important objectives. For instance, if primaries are being chosen from four manipulated variables for two set point objectives, the choices for the first primary would be ranked from one to four. For the second primary, three choices are available if it is assumed that the first primary does not saturate. If the first primary is assumed to be saturated (thus the second choice for the first primary is used), two possibilities remain for the second primary. Their "closed-loop" behavior depends on the second choice for the first primary, and thus they may be ranked differently than in the unsaturated case.

Additional Properties of Static MMCs

In addition to being able to explicitly handle robustness issues, a static MMC should be integrally controllable, be able to tolerate sensor and actuator failures, and permit explicit analysis of its static feasibility when the desired steady state operation has been specified. In this section we will examine each one of these three requirements and how the MMC structure meets them.

Stability properties of MMC derived from steady state information

MMC, like most process controllers, can be characterized as an integrator, $(1/s)\mathbf{I}$, coupled with a compensator, $C(s)$, which controls the plant, represented by the matrix $K(s)$, in a feedback loop as in Figure 4. It is possible to derive closed-loop stability properties from the open-loop steady-state gain matrixes $C(0)$ and $K(0)$, hereafter referred to as C and K (Grosdidier et al., 1985; Campo and Morari, 1990). The steady-state compensator, C, that results from the design methodology of MMC is the inverse of the closed-loop gain matrix, G, between the control outputs (objectives, goals) and the corresponding set of primary manipulated variables (that is, $C=G^{-1}$). Note that G is the upper triangular matrix that results from the decomposition:

$$K = L \cdot D \cdot U \tag{34}$$

where $G=U$, an upper triangular matrix, and L and D are the corresponding lower triangular and diagonal matrices. The product of the controller and the plant, H, is:

$$H = G \cdot C = L \cdot D \cdot U \cdot U^{-1} = \mathbf{L} \cdot \mathbf{D} \tag{35}$$

given perfect modelling. $H = L \cdot D$ is always a lower triangular matrix with 1's on the diagonal, that is, H is positive definite. This proves that given perfect modelling, an MMC controlled system always exhibits *integral controllability* (Grosdidier et al., 1985) or equivalently is *unconditionally stable* (Campo and Morari, 1990). A system is unconditionally stable if $\hat{C} = E \cdot C$ stabilizes the system for all E of the form $E = \alpha I$, $\alpha \epsilon (0,1]$. In other words, the controller gains for each objective may vary

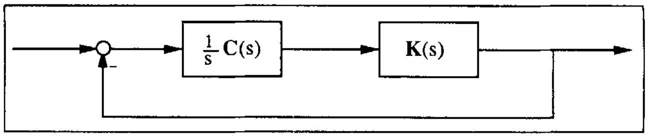

Figure 4. Standard feedback process controller.

up to a constant. While unconditional stability as defined above only allows simultaneous variation of the controller gains, it can be shown that the controller gains for each objective may be tuned independently up to a constant with no loss of stability (see Appendix A for proof). Loss of unconditional stability would only occur under the same conditions as a loss of primary due to robustness considerations, as discussed in the section on selection of primary manipulated variables. The strategy of selecting the primary for a given objective (Criterion 2) in conjunction with the introduction of a second primary manipulated variable (see the next two sections for actuator failure or operational saturation) allows the MMC to maintain unconditional stability for as many of the most important control objectives as the number of unsaturated (or unfailed) actuators. All subsystems made up of the n most important objectives (where n can vary between 1 and all of the objectives) are also *unconditionally stable.*

Tolerance to sensor and actuator failures

The implicit design of many multivariable controllers renders them inoperable in the presence of sensor and/or actuator failures. Several ad hoc strategies (heuristic rules, specific reconfigurations of control systems) have been proposed to deal with such situations, but none of them is a natural component of the design methodology, nor are they always easily understood by the operating personnel. The modular character of an MMC allows the natural treatment of sensor and/or actuator failures within the scope of the same design methodology.

An MMC controller always automatically reconfigures itself to produce an unconditionally stable system. It will always satisfy the maximum possible number of objectives in the order of importance. The reconfiguration is neither ad hoc nor does it require specific antifault design. It is an implicit result of the general design, due in part to a coordinated controller's ability to reject MV-target requests in order to maintain its own objective. Consider the specific cases of sensor and actuator failure.

Sensor Failure. In the case of a sensor failure, corresponding to the ith control objective (ith coordinated controller) the MMC controller could be instructed to react in one of the following two modes, selected and directed by the process operator:

Mode-1 (*Closed-Loop*). MMC is instructed to consider the model for predicting y_i as "perfect", that is, set the feedback signal in that coordinated controller to zero, forcing it to behave as a strictly feedforward controller. In this case the MMC retains unconditional stability by maintaining the control objective y_i to the best of the (blind) model's ability.

Mode-2 (*Open-Loop*). The operator turns the affected ith coordinated controller off, maintaining the rest of the MMC under closed-loop operation. In this case, the primary of the ith coordinated controller will take on its MV-target value (that is, become inactive), thus adding a degree of freedom to the overall control system. The unconditional stability of the more important control outputs is trivially maintained. While the selection of primary variables for the less important coordinated controllers now may not be optimal, nominal unconditional stability is maintained even for these objectives as each coordinated controller automatically adjusts to new closed-loop gains.

Actuator Failure. Loss or saturation of an actuator would result in the loss of a primary in one of the MMC's coordinated controllers, for example, the ith one. In such a case, the second best primary (identified during the design phase, see the section on selection of primary manipulated variables) becomes active as the primary of the ith coordinated controller. This is followed by a re-evaluation of the closed-loop gains for all coordinated controllers with relative priority greater than i (less important), which maintains unconditional stability for the whole system. The unconditional stability of the more important objectives is trivially achieved as the closed-loop gains of their primaries do not change. Closed-loop gains are recalculated in the less important coordinated controllers. The net result is that an extra manipulated variable must become active as a primary in order to replace the lost degree of freedom. If the number of failed actuators exceeds the number of extra manipulated variables, the MMC will start losing control of objectives, beginning with the least important objective.

Operating feasibility of MMCs

Given a desired steady state, it is imperative that the employed controller design methodology can determine whether the desired operating conditions are feasible or not. The structure of MMC provides a direct and explicit answer to this question. Three cases need to be examined, where there are one, zero, or many feasible solutions to be found.

One Solution (*Fully Specified*). In this case there are as many free (not saturated) manipulated variables, N, as there are active objectives, M (that is, $N=M$). An active objective is a set point, or an inequality which is at its equality limit, thus requiring active control to maintain that limit. The controller need only invert an $N\times N$ matrix to find the solution. If a manipulation saturates while trying to reach this solution, N is reduced by 1 for each saturation and the system becomes increasingly over-specified (see below).

Zero Solutions (*Overspecified*). In an overspecified system, $M>N$, so it is not possible to satisfy all M objectives. An MMC controller deals with this situation by relaxing the least important objectives, one by one. Each objective that is relaxed reduces M by 1. When $M=N$, all of the most important objectives that can be satisfied have been satisfied.

Many Solutions (*Underspecified*). In an underspecified system, $M<N$, that is, all of the objectives can be specified without using all of the manipulated variables. MMC leaves those manipulations that are not assigned as primary manipulated variables at their MV-target values.

An MMC controller will always find a feasible solution if one exists. The procedure to determine the feasibility of an operating point is outlined in detail in Appendix A. Briefly, the algorithm iteratively determines which manipulations saturate (decreasing N) and which inequalities are active (increasing M) as it approaches a solution. After M grows to be equal to N, each saturation or newly active constraint results in the least important objective being relaxed. The manipulated variable space is linear and convex so any algorithm which always improves its solution at each step will converge.

Case Study: Shell Standard Control Problem

Consider the heavy oil fractionator of Figure 5 (Prett and Garcia, 1988). The operational control objectives are:

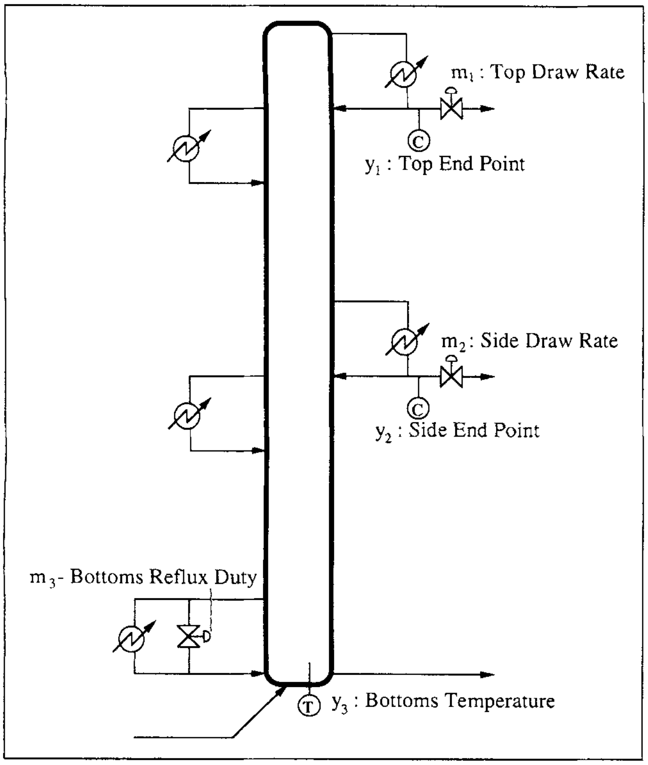

Figure 5. Shell standard control problem.

Garcia, 1988). The operational control objectives are:

Goal-1: Maintain the top product's end point at a given setpoint ($y_1 = y_{1,sp}$).

Goal-2: Maintain the side product's end point at a given setpoint ($y_2 = y_{2,sp}$).

Goal-3: Maximize the steam production by the bottom circulating reflux (maximize bottom's reflux duty, y_4).

Goal-4: The temperature of the bottom's reflux draw should be always above a minimum value ($y_3 \geq y_{3,lo}$).

Goal-5: The top product's end point should be maintained within a range of maximum and minimum values ($y_{1,up} \geq y_1 \geq y_{1,lo}$).

Goal-6: The side product's end point should be maintained within a range of maximum and minimum values ($y_{2,up} \geq y_2 \geq y_{2,lo}$).

In addition, the three manipulated variables, top draw (m_1), side draw (m_2) and bottoms reflux duty (m_3), are constrained to vary within $[-1, 1]$. The original statement of this problem, which is known as the Shell Standard Control Problem, includes specifications on the allowable rates of change of the manipulated variables, and the closed-loop speed of response. These dynamic requirements will not be considered here. We will examine two distinct priority rankings of the six operating goals.

The steady-state model for the Shell Standard control problem is as follows (Prett and Garcia, 1988):

$$[y_1\ y_2\ y_3\ y_4]^T = \boldsymbol{K} \cdot [m_1\ m_2\ m_3]^T \tag{36}$$

where each column of $\boldsymbol{K} = [k_1\ k_2\ k_3]$ may vary between an upper and a lower bound

$$k_{0,j} - e_j \leq k_j \leq k_{0,j} + e_j \quad j = 1, 2, 3 \tag{37}$$

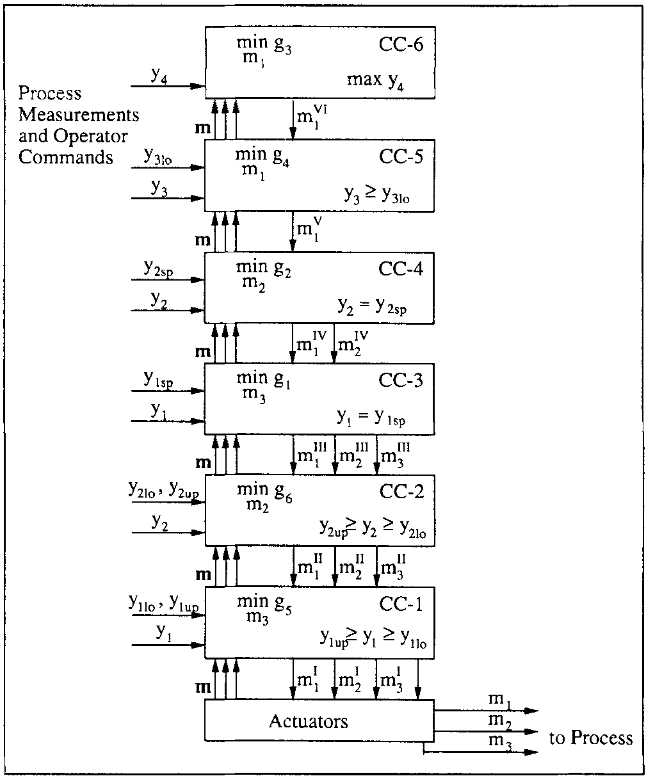

Figure 6. MMC for the shell standard control problem, using the priority ranking 1.

where

$$\boldsymbol{K}_0 = \begin{pmatrix} 4.05 & 1.77 & 5.88 \\ 5.39 & 5.72 & 6.90 \\ 4.38 & 4.42 & 7.20 \\ 0.00 & 0.00 & 1.00 \end{pmatrix}; \ \boldsymbol{E} = \begin{pmatrix} 2.11 & 0.39 & 0.59 \\ 3.29 & 0.57 & 0.89 \\ 3.11 & 0.73 & 1.33 \\ 0.00 & 0.00 & 0.00 \end{pmatrix} \tag{38}$$

MMC configuration for priority ranking 1

Here we assume that Goal-4, the bottom's temperature constraint is less important than the two set point objectives. Goals -5 and -6 (end point constraints) must be more important than Goals -1 and -2, respectively, or they would be pointless as the former pair are trivially satisfied if the latter pair are satisfied. Further, the top draw product is more important than the side draw product. We assume that the production of steam, Goal-3, is the least important objective. Thus the priority ranking is:

Goal-5 > Goal-6 > Goal-1 > Goal-2 > Goal-4 > Goal-3 (PR-1)

The full configuration diagram for Eq. PR-1, containing the results of the design described below, appears in Figure 6. We will number the coordinated controllers according to their importance, rather than according to their goal numbers.

The first coordinated controller (CC-1 for Goal-5) constrains output y_1. From the information in Eq. 38, we can calculate the gain and robustness measure, C_g, for each of the three choices for primary. This is shown in Table 1. For the most

Table 1.

CC-1 Primary	m_1	m_2	m_3
C_g	3.18	1.57	1.22
C.L. Gain	4.05	1.77	5.88

important objective, note that the "closed-loop" gains are equal to the open loop gains. Here we find that the primary with the smallest C_g and the largest "closed-loop" gain is m_3, therefore we choose m_3. Note that the most obvious choice to regulate the top end point, the top draw (m_1), was not chosen, in part because of its poor modelling. This leaves two choices for the primary for the second coordinated controller (CC-2), dealing with output y_2. This is shown in Table 2. Here we see that m_2 has the lower C_g and the higher "closed-loop" gain, so m_2 is the second primary. As discussed in the section on model uncertainty and its impact on the design of robust static MMCs, $C_g=\infty$ signals that the sign of the modeled "closed-loop" gain could be different than the sign of the real "closed-loop" gain for m_1.

Coordinated controllers CC-3 and CC-4 will maintain y_1 and y_2 at their set points, respectively, just as CC-1 and CC-2 ensure that they remain within bounds. Because each pair of set point and constraint controllers cannot be active at the same time, the process of choosing the primaries for CC-3 and CC-4 is the identical to the process for choosing the primaries for CC-1 and CC-2. Thus, not surprisingly, the same primaries, m_3 for CC-3 (affecting y_1) and m_2 for CC-4 (affecting y_2) will be chosen. CC-5 maintains Goal-4, the bottoms temperature constraint ($y_3 \geq y_{3,lo}$). As m_2 and m_3 will be used by some combination of the first four coordinated controllers, only m_1 remains to be the primary for CC-5. This is shown in Table 3. Note that the poor modelling of m_1 for all outputs which results in $C_g=\infty$ does not affect the stability robustness for the more important objectives.

Table 4 shows CC-6, which attempts to maximize steam made in the bottom's reflux, Goal-3, must also have the primary m_1. In calculating the "closed-loop" gain and C_g, we assume that Goal-4 is not at its constraint (otherwise, any MV-target signal from CC-6 would be ignored by CC-5). If the inequality objective of Goal-4 is active, the system is overspecified and the controller cannot satisfy the least important goal, Goal-3.

To maximize steam make we do not actually require a control law mechanism in CC-6 because Goal-3 is a maximization rather than a set point or constraint. The negative "closed-loop" gain indicates that to maximize y_4, we need only set the manipulated variable target for m_1 to a constant -1, the lower limit for m_1.

MMC configuration for priority ranking 2

In this configuration we assume that Goal-4, the bottom's

Table 2.

CC-2 Primary	m_1	m_2
C_g	∞	1.10
C.L. Gain	0.64	3.64

Table 3.

CC-5 Primary	m_1
C_g	∞
C.L. Gain	-5.72

Table 4.

CC-6 Primary	m_1
C_g	2.75
C.L. Gain	-0.65

temperature constraint, is in fact the most important objective. The top product end point is still more important than the side product end point, so all other relative rankings remain the same, that is:

$$\text{Goal-4} > \text{Goal-5} > \text{Goal-6} > \text{Goal-1} > \text{Goal-2} > \text{Goal-3} \quad \text{(PR-2)}$$

The full configuration diagram for Eq. PR-2 appears in Figure 7.

The first coordinated controller, CC-1, is responsible for Goal-4, keeping y_3 above its lower bound. The choices for

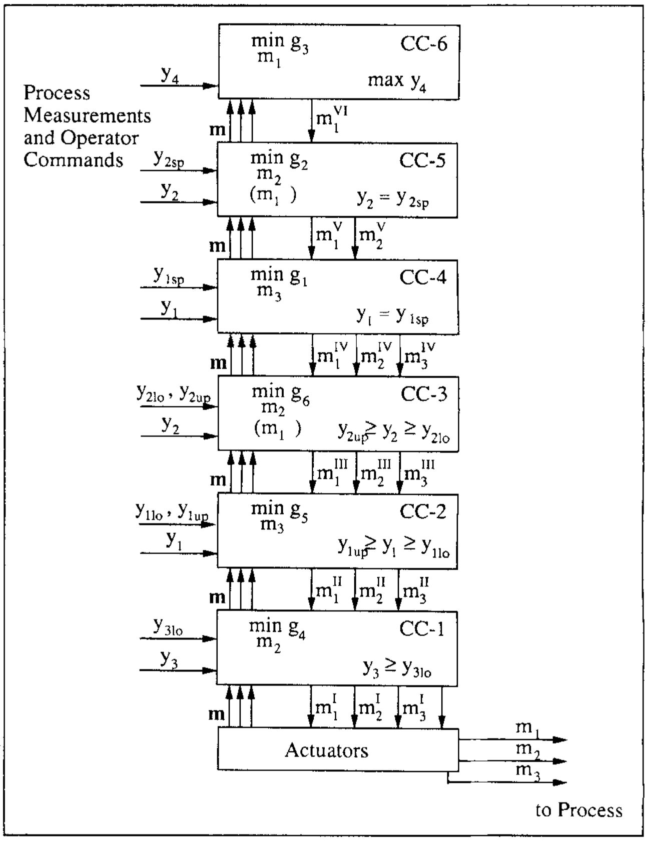

Figure 7. MMC for the shell standard control problem, using the priority ranking 2.

Table 5.

CC-1 Primary	m_1	m_2	m_3
C_g	5.90	1.40	1.45
C.L. Gain	4.38	4.42	7.20

Table 7.

CC-3 Primary	m_1
C_g	4.1
C.L. Gain	1.57

primary are shown in Table 5. Here we have a conflict in the design procedure because our two objectives, maximizing "closed-loop" gain and minimizing C_g, are in conflict. We will carry both options m_2 and m_3 to the next step to see which results in a better design there.

CC-2 keeps y_1 within its bounds. The choices for primary depend on whether the constraint in CC-1 is active (at its constraint) or not. If CC-1 is not active, the choice for CC-2 primary is the same as was the case for the first coordinated controller in Priority Ranking 1, that is, m_3, because CC-1 will not change any MV-target requests as they pass through it. If, on the other hand, CC-1 is active, the choices for primary are shown in Table 6. Of these four possible options, we see that choosing m_2 as the CC-1 primary and m_3 as the CC-2 primary results in the largest "closed-loop" gain and the smallest C_g. Thus m_3 is the choice for primary whether or not CC-1 is active (that is, the inequality is at its equality).

If CC-1 is active, only one choice remains for the primary for CC-3. This is shown in Table 7. If CC-1 is not active, the primary for CC-3 should be m_2, as was the case for the second coordinated controller in Priority Ranking 1. We will make m_2 the first choice for the CC-3 primary and m_1 the second choice. If CC-1 is active, changes to m_2 will be ignored and CC-3 will automatically change to its second choice for primary, m_1.

As was the case in the first priority ranking, the primary assignments for CC-4 and CC-5 are the same as those for CC-2 and CC-3. The design of the least important steam-make optimizer, CC-6, is identical to its design in Priority Ranking 1.

A set of example calculations, using only coordinated controllers 1, 4, and 5 is presented in Appendix B.

Conclusions

Modular Multivariable Control (MMC) is a new multivariable control methodology which, while completely general in its solution of the multivariable problem, allows a greater transparency in multivariable control system design. Its modular structure allows the inherent strengths and weaknesses of the process and its model to be apportioned so that good properties (for example, good robust stability) can be associated with the most important objectives while poor properties are associated with the least important objectives. Inequality constraint objectives are fully incorporated into the design procedure by demanding the specification of each constraint objective's importance relative to all objectives (including set points) and each constraint's desired degree of rigidity (hard or soft).

MMC uses *lexicographic goal programming* to find an optimal solution to the multivariable control problem. Each objective is ranked from most important to least important. The controller finds the solution where as many objectives are satisfied *in the order of importance*, as is possible given the available degrees of freedom. A priori determination of objective weights is not necessary as is the case in sum-of-squares utility function approaches. This simplifies the task of converting engineering specifications and desires to control objectives.

A new measure of robust stability is introduced which has engineering significance. An MMC controller is integrally controllable, allowing on-line tuning, one objective at a time. The hierarchical ranking of objectives leads to robust sensor and actuator failure tolerance. Finally, an MMC controller will always find a feasible solution if one exists. Feasibility analysis of an MMC controlled system is readily accomplished through the solution of a linear program.

Table 6.

	m_2 as CC-1 Primary		m_3 as CC-1 Primary	
CC-2 Primary	m_1	m_3	m_1	m_2
C_g	2.38	1.16	∞	1.75
C.L. Gain	2.30	3.00	0.47	−1.84

Literature Cited

Biegler, L. T., "On Solving the Fundamental Control Problem in the Presence of Uncertainty: A Mathematical Programming Approach," *Shell Process Control Workshop*, D. M. Prett, C. E. Garcia, and B. L. Ramaker, eds., Butterworths, Stoneham, MA (1990).

Brosilow, C. B., "Modular Multivariable Control Applied to the Shell Heavy Oil Fractionator Problem," *Shell Process Control Workshop*, D. M. Prett, C. E. Garcia, and B. L. Ramaker, eds., Butterworths, Stoneham, MA (1990).

Campo, P. J., T. R. Holcomb, M. S. Gelormino, and M. Morari, "Decentralized Control System Design for A Heavy Oil Fractionator—The Shell Control Problem," *Shell Process Control Workshop*, D. M. Prett, C. E. Garcia, and B. L. Ramaker, eds., Butterworths, Stoneham, MA (1990).

Chankong, V., and Y. Y. Haimes, *Multiobjective Decision Making: Theory and Methodology*, p. 302, North-Holland, New York (1983).

Cuthrell, J. E., D. E. Rivera, W. J. Schmidt, and J. A. Vegeais, "Solution to the Shell Standard Control Problem," *Shell Process Control Workshop*, D. M. Prett, C. E. Garcia, and B. L. Ramaker, eds., Butterworths, Stoneham, MA (1990).

Grosdidier, P., M. Morari, and B. R. Holt, "Closed Loop Properties from Steady State Gain Information," *Ind. Eng. Chem. Fundam.*, **24**, 221 (1985).

Holt, B. R., and Z. Lu, "Scheduled Controllers for Robust, Non-Linear Control: Introduction and Application to the Shell Standard Control Problem," *Shell Process Control Workshop*, D. M. Prett, C. E. Garcia, and B. L. Ramaker, eds., Butterworths, Stoneham, MA (1990).

Ignizio, J. P., *Goal Programming and Extensions*, Heath (Lexington Series), Lexington, MA (1976).

Ignizio, J. P., *Linear Programming in Single and Multiple-Objective Systems*, Prentice-Hall, Englewood Cliffs, NJ (1982).

Ignizio, J. P., "Generalized Goal Programming," *Comput. and Ops. Res.*, **4**, 277 (1983).

Ijiri, Y., *Managerial Goals and Accounting for Control*, Rand-McNally, Chicago (1965).
Jääskelainen, V., *Linear Programming and Budgeting*, Petrocelli/Charter, New York (1976).
Kreisselmeier, G., and R. Steinhauser, "Systematic Control Design by Optimizing a Vector Performance Index," *Proc. IFAC Symp. on Computer-Aided Design of Control Systems*, Zurich, p. 113 (1979).
Kreisselmeier, G., and R. Steinhauser, "Application of Vector Performance Optimization to a Robust Control Loop Design for a Fighter Aircraft," *Int. J. of Control*, **7**, 251 (1983).
Manousiouthakis, V., "A Game Theoretic Approach to Robust Controller Synthesis and the Shell Control Problem," *Shell Process Control Workshop*, D. M. Prett, C. E. Garcia, and B. L. Ramaker, eds., Butterworths, Stoneham, MA (1990).
Morari, M., and E. Zafiriou, *Robust Process Control*, Prentice-Hall, Englewood Cliffs, NJ (1989).
Papadimitriou, C. H., and K. Steiglitz, *Combinatorial Optimization*, p. 166, Prentice-Hall, Englewood Cliffs, NJ (1982).
Popiel, L., C. Brosilow, and T. Matsko, "Coordinated Control," *Chemical Process Control—CPCIII*, M. Morari and T. J. McAvoy, eds., Elsevier Science, New York (1986).
Prett, D. M., and C. E. Garcia, *Fundamental Process Control*, Butterworths, Stoneham, MA (1988).
Prett, D. M., and M. Morari, *The Shell Process Control Workshop*, p. 355, Butterworths, Boston (1987).
Scales, L. E., *Introduction to Non-Linear Optimization*, p. 142, Springer-Verlag, New York (1985).
Skogestad, S., and M. Morari, "Control Configurations for Distillation Columns," *AIChE J.*, **33**, 1620 (1987).
Skogestad, S., and M. Morari, "Effect of Disturbance Directions on Control Performance," *Ind. Eng. Chem. Research*, **26**, 2029 (1987).
Zafiriou, E., "An Operator Control Theory Approach to the Shell Standard Control Problem," *The Second Shell Process Control Workshop*, D. M. Prett, C. E. Garcia, and B. L. Ramaker, eds., Butterworths, Stoneham, MA (1990).

Appendix A

The following proofs and procedures are referred to in the text:
Proof of identity as shown previously:

$$G_i = \frac{\mathrm{Det}(\boldsymbol{K}^i_{p_1 \ldots p_{i-1} p_i})}{\mathrm{Det}(\boldsymbol{K}^{i-1}_{p_1 \ldots p_{i-1}})} \tag{30}$$

Proof:
Divide $\boldsymbol{K}^i_{p_1 \ldots p_{i-1} p_i}$ into block submatrices such that:

$$\boldsymbol{K}^i_{p_1 \ldots p_i} = \begin{pmatrix} k_{1p_1} & \ldots & k_{1p_{i-1}} & k_{1p_i} \\ \cdot & \ldots & \cdot & \cdot \\ k_{i-1p_1} & \ldots & k_{i-1p_{i-1}} & k_{i-1p_i} \\ k_{ip_1} & \ldots & k_{ip_{i-1}} & k_{ip_i} \end{pmatrix} = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{D} \end{pmatrix}, \tag{A1}$$

$$\boldsymbol{A}_{(i-1)\times(i-1)},\ \boldsymbol{B}_{(i-1)\times 1},\ \boldsymbol{C}_{1\times(i-1)},\ \boldsymbol{D}_{1\times 1}$$

$$G_i = \boldsymbol{D} - \boldsymbol{C}\boldsymbol{A}^{-1}\boldsymbol{B} \quad \text{(as defined in Eq. 29)} \tag{A2}$$

$$= \mathrm{Det}(\boldsymbol{D} - \boldsymbol{C}\boldsymbol{A}^{-1}\boldsymbol{B}) \quad (G_i \text{ is a scalar})$$

$$= \frac{\mathrm{Det}(\boldsymbol{K}^i_{p_1 \ldots p_i})}{\mathrm{Det}(\boldsymbol{A})} \quad \text{(by the identity below)}$$

For any matrix $\boldsymbol{Q}$ where $\boldsymbol{A}^{-1}$ exists,

$$\boldsymbol{Q} = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{D} \end{pmatrix} = \begin{pmatrix} \boldsymbol{I} & 0 \\ \boldsymbol{C}\boldsymbol{A}^{-1} & \boldsymbol{I} \end{pmatrix} \begin{pmatrix} \boldsymbol{A} & 0 \\ 0 & (\boldsymbol{D} - \boldsymbol{C}\boldsymbol{A}^{-1}\boldsymbol{B}) \end{pmatrix} \begin{pmatrix} \boldsymbol{I} & \boldsymbol{A}^{-1}\boldsymbol{B} \\ 0 & \boldsymbol{I} \end{pmatrix} \tag{A3}$$

therefore, $\mathrm{Det}(\boldsymbol{Q}) = \mathrm{Det}(\boldsymbol{A})\mathrm{Det}(\boldsymbol{D} - \boldsymbol{C}\boldsymbol{A}^{-1}\boldsymbol{B})$

Proof of Theorem 1 as shown previously:
Define $D_A \triangleq \det(A)$;

$C_{A,a} \triangleq$ cofactor of A with respect to a, an element of the matrix A.

$$\frac{\partial D_A}{\partial a} = C_{A,a} \tag{A4}$$

$C_{A,a}$ is not a function of a. Therefore, D_A is a linear function of a.

Therefore, maximum and minimum values of D_A under variation of any element a must be found at one or the other of the extreme values of a.

Extension to include input and output correlations. Let $a = (a_1, a_2, \ldots, a_m)$ be a set of elements of A, all from the same row or column, which all vary correlated with ϵ, that is:

$$a_i = a_{i,o} + \alpha_i \cdot \epsilon \quad i = 1, 2, \ldots, m, \quad \text{where } |\epsilon| \le 1 \tag{A5}$$

$$\frac{\partial D_A}{\partial \epsilon} = \alpha_1 \frac{\partial D_A}{\partial a_1} + \alpha_2 \frac{\partial D_A}{\partial a_2} + \ldots + \alpha_m \cdot \frac{\partial D_A}{\partial a_m}$$
$$= \alpha_1 \cdot C_{A,a_1} + \alpha_2 \cdot C_{A,a_2} + \ldots + \alpha_m \cdot C_{A,a_m} \tag{A6}$$

As in Eq. A4, the right-hand side of Eq. A6 is not a function of ϵ, because the cofactors in Eq. A6 are not a function of any of the elements in a. Thus D_A is a linear function of ϵ and the maximum and minimum values of D_A are found at the upper or lower extreme values of ϵ. The linear dependence that this proof depends on does not exist for sets of correlated elements that are not in the same row or column.
Proof of Theorem 2 as shown previously:

$$G_i = \frac{D_A}{D_{A'}} \quad \text{(from Eq. 32)} \tag{A7}$$

Allowing all elements a of A to vary between an upper and a lower bound, independently, it is assumed that: (i) D_A cannot change sign, and (ii) $D_{A'}$ cannot change sign. For any matrix element a^* which is in A but not in A':

$$\frac{\partial G_i}{\partial a^*} = \frac{C_{A,a^*}}{D_{A'}} \tag{A8}$$

which is not a function of a^*. Therefore, G_i is a linear function of a^*. For any parameter a which is an element of both A and $\mathbf{A}'$:

$$\frac{\partial G_i}{\partial a} = \frac{C_{A,a} D_{A,'} - C_{A,a} D_A}{D^2_{A'}} \tag{A9}$$

$$=\frac{C_{A,a}(a\cdot C_{A,',a}+\beta_1)-C_{A',a}(a\cdot C_{A,a}+\beta_2)}{D_{A'}^2}$$

$$=\frac{C_{A,a}\cdot\beta_1-C_{A,',a}\cdot\beta_2}{D_{A'}^2} \tag{A10}$$

where β_1 and β_2 are not functions of a.

The numerator of Eq. A10 is not a function of a and the denominator cannot change sign (by assumption, above) so $(\partial G_i)/(\partial a)$ cannot change sign, for all values of a.

Therefore all maxima and minima of G_i must occur at a corner, that is, at extreme constrained values, in the feasible space of the elements of A.

Extension to include input and output correlations. Let $a=(a_1, a_2, \ldots, a_m)$ be a set of elements of A, all from the same row or column, which all vary correlated with ϵ, as defined in Eq. A5:

$$\frac{\partial G_i}{\partial \epsilon}=\frac{\frac{\partial D_A}{\partial \epsilon}\cdot D_{A'}-\frac{\partial D_{A'}}{\partial \epsilon}\cdot D_A}{D_{A'}^2} \tag{A11}$$

$$=\frac{\frac{\partial D_A}{\partial \epsilon}\left(\epsilon\cdot\frac{\partial D_{A'}}{\partial \epsilon}+\beta_3\right)-\frac{\partial D_{A'}}{\partial \epsilon}\left(\epsilon\cdot\frac{\partial D_A}{\partial \epsilon}+\beta_4\right)}{D_{A'}^2}$$

$$=\frac{\frac{\partial D_A}{\partial \epsilon}\cdot\beta_3-\frac{\partial D_{A'}}{\partial \epsilon}\cdot\beta_4}{D_{A'}^2} \tag{A12}$$

where β_3 and β_4 are not functions of ϵ.

The expansion of the partial derivatives on the right-hand side of Eq. A11 is as defined in Eqs. A5 and A6; neither $(\partial D_A)/(\partial\epsilon)$ nor $(\partial D_{A'})/(\partial\epsilon)$ are functions of ϵ for row and column correlations. The rest of the proof follows as above. As was the case in theorem 1, the proof breaks down for correlations that are not along rows or columns because $(\partial D_A)/(\partial\epsilon)$ and $(\partial D_{A'})/(\partial\epsilon)$ become functions of ϵ, which allows $(\partial G_i)/(\partial\epsilon)$ to change sign.

Proof of Theorem 3 as shown previously:

(*i*) *Upper Bound of* G_i. Recall Eq. 32, determining the value of G_i, that is:

$$G_i=\frac{\det(A)}{\det(A')} \tag{32}$$

The elements of A, which include all those of A', are allowed to vary between upper and lower bounds. Therefore, all attainable values of G_i, including $G_{i,\max}$ and $G_{i,\min}$, are subject to the inequality:

$$\frac{\det(A)_{\min}}{\det(A')_{\max}}\le G_i\le\frac{\det(A)_{\max}}{\det(A')_{\min}} \tag{A12}$$

Therefore:

$$C_g=\frac{G_{i,\max}}{G_{i,\min}}\le\frac{\det(A)_{\max}\det(A')_{\max}}{\det(A)_{\min}\det(A')_{\min}} \tag{A13}$$

(*ii*) *Lower Bound of* G_i. All possible values of G_i must satisfy:

$$G_{i,\min}\le G_i\le G_{i,\max} \tag{A14}$$

If $G_i(\det(A)_{\max})$ is the value for G_i when the values of the elements of A are set such that the determinant of A is maximized:

$$\frac{\det(A)_{\max}}{\det(A')_{\max}}\le G_i(\det(A)_{\max})\le G_{i,\max} \tag{A15}$$

$$\frac{\det(A)_{\min}}{\det(A')_{\min}}\le G_i(\det(A')_{\min})\le G_{i,\max} \tag{A16}$$

$$G_{i,\min}\le G_i(\det(A')_{\max})\le\frac{\det(A)_{\max}}{\det(A')_{\max}} \tag{A17}$$

$$G_{i,\min}\le G_i(\det(A)_{\min})\le\frac{\det(A)_{\min}}{\det(A')_{\min}} \tag{A18}$$

define Q as one of the set

$$\left\{\frac{[\det(A)]_{\max}}{[\det(A')]_{\max}},\ \frac{[\det(A)]_{\min}}{[\det(A')]_{\min}},\ G_{i,\text{nominal}}\right\}$$

$$C_g=\frac{G_{i,\max}}{G_{i,\min}}\ge\frac{Q_{\max}}{Q_{\min}} \tag{A19}$$

Iterative method for finding maximum or minimum determinants and gains

A determinant is a linear function of its elements, as was shown in theorem 1. In a linear space, any maximum (minimum) point is the global maximum (minimum). The following theorem, which shows the same for "closed-loop" gains, is necessary for the procedure below to guarantee finding the global gain maximum and minimum.

Theorem A1: Any maximum (*minimum*) *gain is a global maximum* (*minimum*) *gain*

Proof: (by contradiction)

Consider a line r drawn in the feasible element space of the elements of A from the global maximum (minimum) M to a local maximum (minimum) m. From the theorem above, both M and m must be at a corner in the feasible space. Now consider the graph of the gain vs. distance along r. Somewhere on this graph is a point P which is a minimum (maximum). By the convexity of the linear constraint space, this minimum (maximum) cannot be a corner in the feasible space. This is a contradiction of Theorem 1.

Calculating Maximum and Minimum Determinants. The determinant of a matrix is a linear function of any element of that matrix. Therefore if the elements of that matrix are allowed to vary independently between an upper and a lower bound, the unique maximum and the unique minimum value of the determinant must occur at a corner of the convex feasible space of the elements of the matrix, by the proofs above. This allows us to use this simple LP solution method to find the possible determinant extrema.

Let the independent variations of the elements of a matrix $A=[a_{ij}]$ by described by a nominal matrix $K=[k_{ij}]$ and a matrix of possible errors $E=[e_{ij}]$ such that:

$$k_{ij}-e_{ij}\leq a_{ij}\leq k_{ij}+e_{ij}$$

The procedure is as follows:

- Calculate $R=[r_{ij}]$, s.t. $r_{ij}=C_{K,k_{ij}}$.
- Calculate $S=[s_{ij}]$, s.t. $s_{ij}=\text{sign}(r_{ij})$.
- Calculate $T=[t_{ij}]$, s.t. $t_{ij}=k_{ij}+s_{ij}e_{ij}$ (maximum) $k_{ij}-s_{ij}e_{ij}$ (minimum)
- Calculate $R^*=[r'_{ij}]$, s.t. $r'_{ij}=C_{T,t_{ij}}$.
- Calculate $S^*=[s'_{ij}]$, s.t. $s'_{ij}=\text{sign}(r'_{ij})$.
- If $S^*\neq S$ then let $S=S^*$; go to 3.
- $\det(A)_{\max}$ (or $\det(A)_{\min}$) $=\det(T)$

Find maximum and minimum determinants first. If the maximum and minimum determinants are not the same sign, then C_g is infinity. If C_g is finite, use the following similar procedure to find the maximum and minimum possible "closed-loop" gains:

Calculating Maximum and Minimum Closed Loop Gains. Define $F'\cong$ largest principal minor of any matrix F

The "closed-loop" gain $G_i=D_A/D_{A'}$, where A, K, and E are as defined above. A, K and E are $n\times n$ matrices.

- Calculate R

$$=[r_{ij}],\ \text{s.t.}\ r_{ij}=\frac{C_{K,k_{ij}}D_{K'}-C_{K',k'_{ij}}D_K}{D^2_{K'}},\quad i<n \text{ and } j<n$$

$$r_{ij}=\frac{C_{K,k_{ij}}}{D_{K'}},\quad i=n \text{ or } j=n$$

- Calculate $S=[s_{ij}]$, s.t. $s_{ij}=\text{sign}(r_{ij})$.
- Calculate $T=[t_{ij}]$, s.t. $t_{ij}=k_{ij}+s_{ij}e_{ij}$ (maximum) $k_{ij}-s_{ij}e_{ij}$ (minimum)
- Calculate $R^*=[r'_{ij}]$, s.t. $r'_{ij}=\dfrac{C_{T,t_{ij}}D_{T'}-C_{T',t'_{ij}}D_T}{D^2_{T'}}$,

$$i<n \text{ and } j<n$$

$$r'_{ij}=\frac{C_{T,t_{ij}}}{D_{T'}},\quad i=n \text{ or } j=n$$

- Calculate $S^*=[s'_{ij}]$, s.t. $s'_{ij}=\text{sign}(r'_{ij})$.
- If $S^*\neq S$ then let $S=S^*$; go to 3.
- $\max(G_i)$ [or $\min(G_i)$] $=D_T/D_{T'}$.

Having found the maximum and minimum values of the closed-loop gain, and knowing the G_i does not change sign, the value of the robustness measure or robustness is given by:

$$C_g(G_i)=\frac{\max|G_i|}{\min|G_i|}\tag{26}$$

which allows for G_i to be positive or negative.

This algorithm is a linear programming algorithm similar to the simplex algorithm with no slack variables. It is guaranteed to find the correct solution. Pathologically, it could visit n^2 (B is $n\times n$) possible solutions while finding the maximum and minimum determinant and all 2^{n^2} possible solutions in its search for the maximum and the minimum gain. However, such algorithms have proven to be much more efficient in practice than their pathological complexity suggests (Papadimitriou et al., 1982).

Theorems 1 and A2, along with the convexity of a linear constraint space, are sufficient to allow us to use the algorithm outlined above to find the maximum and minimum gain (Scales, 1985).

Proof of Stable Independent Tuning. The matrix H is positive definite. Changing the controller gain for any coordinated controller governed by H by a factor α will change the determinant of H by the same factor α. As long as the sign of α remains positive, H will remain positive definite and the system will remain unconditionally stable.

Feasibility Analysis Solution Algorithm. This algorithm answers the question "Is an output vector y_e of equality constraints feasible subject to a vector of inequality constraints y_i?"

Consider a system characterized by a vector of manipulated variables m, a vector of outputs y, and a matrix of nominal open loop gains K to relate them. Each element of K is scaled such that the elements m_i of m can only assume values in the range $[-1, 1]$. The maximum possible magnitudes of the possible parameter errors in K are expressed, element by element, in the matrix E. y_e and y_i specify complementary subsets of the vector y.

1. Define y_a, the vector of active constraints (equality and inequality) and m_a, the vector of primaries for y_a. Set $y_a=y_e$ and set m_a accordingly.

2. Solve $m_a=K_a^{-1}\cdot y_a$, (where K_a is the appropriate submatrix of K to relate the manipulated variables and outputs specified in m_a and y_a, respectively).

3. Robust Stability check: if $\min|\det(K_a)|=0$, there is a robustness failure in the controller. The failing primary can be found by successive examination of C_g for each primary. The failing primary will have a gain condition measure of infinity. After determining m_j, the failing primary:

(i) Replace y_a with $y_a+K_j\cdot(m_{aj}/|m_{aj}|)$ (primary saturates in the wrong direction)

(ii) Replace the failed primary with the new primary in m_a. Adjust K_a accordingly.

(iii) Go to 2.

4. For each equality or inequality constraint in order of importance:

4a. Equality or already active inequality constraint (that is, in y_a):

(i) Check solution for the value of the primary.

(ii) If $|m_{aj}|>1$ constraint saturation has occurred. Otherwise continue with 4.

(iii) Replace y_a with $y_a-K_j\cdot(m_{aj}/|m_{aj}|)$ (primary saturates in the correct direction)

(iv) Replace the saturated primary with the new primary in m_a. Adjust K_a accordingly.

(v) Go to 2.

4b. New inequality constraint (that is, not in y_a):

(i) If the solution m_a causes the constraint to be violated the constraint is now active. Otherwise, continue with 4.

(ii) Add the new constraint to y_a (that is, the vector grows in length). Add its primary to m_a. Adjust K_a accordingly.

(iii) Go to 2.

5. If steps 1 through 4 have been completed without running out of unsaturated manipulated variables to use as primaries, then a feasible solution m_a exists for the problem y_e. If this is not the case then the constraint at which the procedure ran out of manipulated variables marks the dividing line between those constraints which would be satisfied and those that would not.

Notes:

- The manipulated variables which are not specified as primaries will in general have nonzero MV-target values. This does not affect the determination of whether a state is feasible or not but it does affect the number of manipulated variables that would be removed from their MV-targets (that is, become active) to achieve that state. To include the effect of MV-targets, change the equation in step 1 to:

$$y_a = y_e - K_{mvt} \cdot m_{mvt}$$
$$y_i = y_i - K'_{mvt} \cdot m_{mvt}$$

where K_{mvt} relates the elements of y_a with the MV-target variables in m_{mvt} and K'_{mvt} relates y_i with m_{mvt}. When an unused manipulated variable m_k becomes a primary (steps 3, 4a and 4b) the effect of its MV-target must also be removed:

Replace y_a with $y_a + K_k \cdot m_{mvt,k}$

Replace y_i with $y_i + K'_k \cdot m_{mvt,k}$

- The stability robustness check (step 3) assumes that all modelled "closed-loop" gains that could change will do so. Failing step 3 neither guarantees that robustness failure could occur, not that it would occur. Removing step 3 makes the above procedure a verification of purely nominal feasibility.

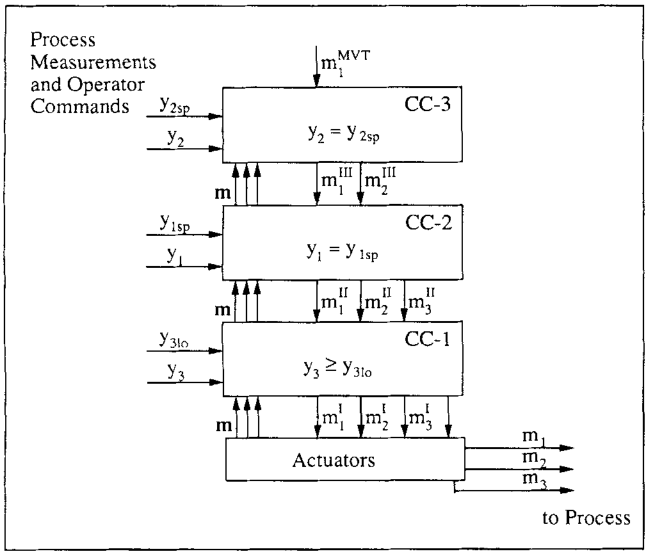

Figure B1. Hierarchy of coordinated controllers in the MMC of the case study in Appendix B.

Appendix B: Numerical Example of the Static MMC

The purpose of this appendix is to illustrate the flow of numerical calculations carried out by the static Coordinated Controllers of an MMC. The example is drawn from the Shell Standard Control Problem (Prett and Morari, 1987) whose process is described by the following static model:

$$y_1 = 4.1 \cdot m_1 + 1.8 \cdot m_2 + 5.9 \cdot m_3 \quad \text{(B1)}$$

$$y_2 = 5.4 \cdot m_1 + 5.7 \cdot m_2 + 6.9 \cdot m_3 \quad \text{(B2)}$$

$$y_3 = 4.4 \cdot m_1 + 4.4 \cdot m_2 + 7.2 \cdot m_3 \quad \text{(B3)}$$

The initial state is described by:

$$m_i = 0 \quad i = 1, 2, 3 \quad \text{and} \quad y_i = 0 \quad i = 1, 2, 3$$

The following operational constraints also apply:

$$1 \geq m_i \geq -1 \quad i = 1, 2, 3 \quad \text{(B4)}$$

Set Point Tracking

Consider a set point change of y_1 from 0 to -2. Figure B1 indicates the hierarchy of the three coordinated controllers and the associated control objectives, that is:

CC-I with control objective $y_3 \geq y_{3,lo} = -2$

CC-II with control objective $y_1 = y_{1,sp} = -2$

CC-III with control objective $y_2 = y_{2,sp} = 0$

Step A: Coordinated Controller CC-I. Let m_2 be the primary manipulated variable, selected to satisfy the $y_3 \geq -2$ control objective. From Eq. B3 and constraint Eq. B4 it is easy to see that the design requirements of CC-I are:

$$y_3 = 4.4 \cdot m_1^I + 4.4 \cdot m_2^I + 7.2 \cdot m_3^I \geq -2$$
$$-1 \leq m_2^I \leq 1 \quad \text{(B5)}$$

From Eq. B5 we take

$$m_2^I \geq \frac{1}{4.4} [-2 - 4.4 \cdot m_1^I - 7.2 \cdot m_3^I] \quad \text{(B6)}$$

With the initial MV-target values (provided by CC-II), that is

$$m_i^I = m_i^{II} (=0) \quad i = 1, 2, 3$$

we can see that condition Eq. B6 is trivially satisfied. Therefore, CC-I does not alter the m_2^I value and passes on to CC-II the condition Eq. B6.

Step B: Coordinated Controller CC-II. The control objective is $y_1 = y_{1,sp} = -2$, and we assume that the associated primary manipulation has been selected to be m_3. Therefore, the design requirements for CC-II are:

$$y_1 = 4.1 \cdot m_1^{II} + 1.8 \cdot m_2^{II} + 5.9 \cdot m_3^{II} = -2 \quad (B7)$$

$$1 \geq m_3^{II} \geq -1$$

$$1 \geq m_2^{II} \geq \frac{1}{4.4} \cdot [-2 - 4.4 \cdot m_1^{II} - 7.2 \cdot m_3^{II}] \quad (B8)$$

The last condition on m_2^{II} signifies the fact that the objective of CC-I is still maintained. Equation B7 yields the new MV-target for m_3, that is:

$$m_3^{II} = \frac{1}{5.9} \cdot [-2 - 4.1 \cdot m_1^{II} - 1.8 \cdot m_2^{II}] \quad (B9)$$

From Figure B1 we notice that CC-II receives the MV-target values from CC-III, which initially are:

$$m_1^{II} = m_1^{III} (=0) \quad \text{and} \quad m_2^{II} = m_2^{III} (=0)$$

Using these values, Eq. B9 yields $m_3^{II} = -0.34$ and in turn, condition Eq. B8 yields

$$m_2^{II} \geq \frac{1}{4.4} \cdot [-2 - 4.1(0) - 1.8(-0.34)] = 0.102$$

Clearly the initial value of $m_2^{II} = 0$ does not satisfy the above condition. Therefore, the MV-target for m_2 is adjusted to satisfy objective $y_3 \geq -2$ as an equality, that is:

$$m_2^{I} = m_2^{II} = \frac{1}{4.4} \cdot [-2 - 4.4 \cdot m_1^{II} - 7.2 \cdot m_3^{II}] \quad (B10)$$

Solving Eq. B9 and Eq. B10 simultaneously, with $m_1^{II} = 0$, we find:

$$m_2^{II} = 0.20 \quad m_3^{II} = -0.40$$

Step C: Coordinated Controller CC-III. The control objective is $y_2 = y_{2,sp} = 0$, and we assume that the associated primary manipulation has been selected to be m_2. Therefore, the design requirements for CC-III are

$$y_2 = 5.4 \cdot m_1^{III} + 5.7 \cdot m_2^{III} + 6.9 \cdot m_3^{III} = 0 \quad (B11)$$

$$m_3^{III} = \frac{1}{5.9} \cdot [-2 - 4.1 \cdot m_1^{III} - 1.8 \cdot m_2^{III}] \quad (B12)$$

$$1 \geq m_3^{III} \geq -1$$

$$1 \geq m_2^{III} \geq \frac{1}{4.4} \cdot [-2 - 4.4 \cdot m_1^{III} - 7.2 \cdot m_3^{III}] \quad (B13)$$

Conditions Eqs. B12 and B13 indicate that the objectives of both CC-I and CC-II are maintained. Equation B11 yields the new MV-target for m_2, that is:

$$m_2^{III} = \frac{1}{5.7} \cdot [0 - 5.4 \cdot m_1^{III} - 6.9 \cdot m_3^{III}] \quad (B14)$$

Solving Eqs. B12 and B14 simultaneously with $m_1^{III} = m_1^{MVT} = 0$:

$$m_2^{III} = 0.65 \quad m_3^{II} = -0.54$$

The choice of $m_1^{MVT} = 0$ reflects the desire of the operator to keep $m_1 = 0$ if it is not needed to satisfy any of the objectives. The solution obeys the constraint, Eq. B13, that is:

$$m_2^{III} (=0.65) \geq \frac{1}{4.4} \cdot [-2 - 4.4 \cdot (0) - 7.2 \cdot (-0.54)] = 0.43$$

so the solution is viable. Had any of the inequalities constraining m_2^{III} or m_3^{III} been violated, we would have attempted to find a solution using m_1 as the primary manipulation for CC-III. The final results satisfy the initial set of objectives:

$$y_1 = -2.00 \qquad y_2 = 0.00 \quad y_3 = -1.03$$

$$m_1 (= m_1^{MVT}) = 0 \quad m_2 = 0.65 \quad m_3 = -0.54$$

Disturbance Rejection

Consider the Shell Problem with a disturbance of -4 added to y_3, that is:

$$y_1 = 4.1 \cdot m_1 + 1.8 \cdot m_2 + 5.9 \cdot m_3 \quad (B15)$$

$$y_2 = 5.4 \cdot m_1 + 5.7 \cdot m_2 + 6.9 \cdot m_3 \quad (B16)$$

$$y_3 = 4.4 \cdot m_1 + 4.4 \cdot m_2 + 7.2 \cdot m_3 - 4 \quad (B17)$$

and a *nonzero* initial state described by:

$$m_i = 0 \quad i = 1,2,3$$

$$y_i = 0 \quad i = 1,2$$

$$y_3 = -4$$

The following operational constraints still apply:

$$1 \geq m_i \geq -1 \quad i = 1,2,3 \quad (B18)$$

The controller objectives are to maintain the variables y_1 and y_2 at the set points $y_{1,sp} = 0$ and $y_{2,sp} = 0$ and to keep y_3 above its lower limit of -2. As in the previous example, Figure B1 indicates the hierarchy of the three coordinated controllers and the associated control objectives, that is:

CC-I with control objective $y_3 \geq y_{3,lo} = -2$

CC-II with control objective $y_1 = y_{1,sp} = 0$

CC-III with control objective $y_2 = y_{2,sp} = 0$

Step A. Coordinated Controller CC-I. Let m_2 be the primary manipulated variable, selected to satisfy the $y_3 \geq -2$ control objective. From Eq. B17 and constraint Eq. B18 it is easy to see that the design requirements of CC-I are:

$$y_3 = 4.4\cdot m_1^{I} + 4.4\cdot m_2^{I} + 7.2\cdot m_3^{I} - 4 \geq -2$$

$$1 \geq m_2^{I} \geq -1 \tag{B19}$$

From Eq. B19 we take:

$$m_2^{I} \geq \frac{1}{4.4}\,[2 - 4.4\cdot m_1^{I} - 7.2\cdot m_3^{I}] \tag{B20}$$

With the initial MV-target values (provided by CC-II), that is:

$$m_i^{I} = m_i^{II}\,(=0) \quad i = 1,2,3$$

we see that the constraint, Eq. B20 becomes:

$$m_2^{I} \geq \frac{1}{4.4}\,[2 - 4.4\cdot(0) - 7.2\cdot(0)]$$

Clearly the constraint is violated by the MV-target $m_2^{I} = 0$. Therefore the primary manipulated variable, m_2 is adjusted to satisfy the inequality Eq. B20 as an equality, that is:

$$m_2^{I} = \frac{1}{4.4}\cdot[2 - 4.4\cdot m_1^{I} - 7.2\cdot m_3^{I}] \tag{B21}$$

Solving Eq. B21 with $m_1^{I} = 0$ and $m_3^{I} = 0$, we find:

$$m_2^{I} = 0.45$$

Step B. Coordinated Controller CC-II. The control objective is $y_1 = y_{1,sp} = 0$, and we assume that the associated primary manipulation has been selected to be m_3. Therefore, the design requirements for CC-II are:

$$y_1 = 4.1\cdot m_1^{II} + 1.8\cdot m_2^{II} + 5.9\cdot m_3^{II} = 0 \tag{B22}$$

$$1 \geq m_3^{II} \geq -1$$

$$1 \geq m_2^{II} \geq \frac{1}{4.4}\cdot[2 - 4.4\cdot m_1^{II} - 7.2\cdot m_3^{II}] \tag{B23}$$

The last condition on m_2^{II} signifies the fact that the objective of CC-I is still maintained. Equation B22 yields the new MV-target for m_3, that is:

$$m_3^{II} = \frac{1}{5.9}\cdot[0 - 4.1\cdot m_1^{II} - 1.8\cdot m_2^{II}] \tag{B24}$$

From Figure B1 we notice that CC-II receives the MV-target values from CC-III, which initially are

$$m_1^{II} = m_1^{III}(=0) \quad \text{and} \quad m_2^{II} = m_2^{III}(=0)$$

Using these values, Eq. B24 yields $m_3^{II} = 0$. However, the constraint on m_2^{II}, Eq. B23, will clearly be violated, that is:

$$m_2^{II} \geq \frac{1}{4.4}\cdot[2 - 4.1(0) - 1.8(0)] = 0.455$$

Therefore, the MV-target for m_2 is adjusted to satisfy objective $y_3 \geq -2$ as an equality, that is:

$$m_2^{I} = m_2^{II} = \frac{1}{4.4}\cdot[2 - 4.4\cdot m_1^{II} - 7.2\cdot m_3^{II}] \tag{B25}$$

Solving Eq. B24 into Eq. B25 simultaneously, with $m_1^{II} = 0$, we find:

$$m_2^{II} = 0.90 \quad m_3^{II} = -0.27$$

Step C. Coordinated Controller CC-III. The control objective is $y_2 = y_{2,sp} = 0$, and we assume that the associated primary manipulation has been selected to be m_2. Therefore, the design requirements for CC-III are:

$$y_2 = 5.4\cdot m_1^{III} + 5.7\cdot m_2^{III} + 6.9\cdot m_3^{III} = 0 \tag{B26}$$

$$m_3^{III} = \frac{1}{5.9}\cdot[0 - 4.1\cdot m_1^{III} - 1.8\cdot m_2^{III}] \tag{B27}$$

$$1 \geq m_3^{III} \geq -1$$

$$1 \geq m_2^{III} \geq \frac{1}{4.4}\cdot[2 - 4.4\cdot m_1^{III} - 7.2\cdot m_3^{III}] \tag{B28}$$

Conditions Eq. B27 and B28 indicate that the objectives of both CC-I and CC-II are maintained. Equation B26 yields the new MV-target for m_2, that is:

$$m_2^{III} = \frac{1}{5.7}\cdot[0 - 5.4\cdot m_1^{III} - 6.9\cdot m_3^{III}] \tag{B29}$$

Solving Eqs. B27 and B29 simultaneously with $m_1^{III} = m_1^{MVT} = 0$:

$$m_2^{III} = 0 \quad m_3^{III} = 0$$

which violates Eq. B28 the $y_3 \geq -2$ constraint, that is:

$$m_2^{III} \geq \frac{1}{4.4}\cdot[2 - 4.1(0) - 1.8(0)] = 0.455$$

Because the constraint is active, we adjust Eq. B28 to be an equality, that is:

$$m_2^{III} = \frac{1}{4.4}\cdot[2 - 4.4\cdot m_1^{III} - 7.2\cdot m_3^{III}] \tag{B30}$$

Further, as m_2 is being used to satisfy the $y_3 \geq -2$ inequality, we will have to use the one remaining degree of freedom, m_1, to satisfy the $y_2 = y_{2,sp}$ objective. Equation B26 yields a new MV-target for m_1, that is:

$$m_1^{III} = \frac{1}{5.4}\cdot[0 - 5.7\cdot m_2^{III} - 6.9\cdot m_3^{III}] \tag{B31}$$

Solving Eqs. B27, B30, and B31 simultaneously:

$$m_1^{\mathrm{III}} = -2.08 \quad m_2^{\mathrm{III}} = 1.33 \quad m_3^{\mathrm{III}} = 0.36$$

According to Eq. B18, the process input constraints, this solution is not feasible (manipulated variables saturate at ± 1). As we have no more degrees of freedom to use, we conclude that we are unable to satisfy the third, least important objective, $y_2 = y_{2,sp}$. We will let m_1 saturate in the direction which decreases the deviation from the $y_2 = y_{2,sp}$ greatest, that is, at -1, and solve Eqs. B27 and B30 (representing the first and second objectives). The final results indicate that the first two objectives ($y_3 \geq -2$ and $y_1 = 0$), but not the third objective ($y_2 = 0$) are satisfied:

$$y_1 = 0.00 \quad y_2 = 1.69 \quad y_3 = -2.00$$
$$m_1 = -1.00 \quad m_2 = 0.36 \quad m_3 = 0.64$$

Manuscript received Dec. 30, 1991, and revision received May 11, 1992.