15th Triennial World Congress, Barcelona, Spain

DIRECT MODEL REFERENCE ADAPTIVE CONTROL AND SATURATION CONSTRAINTS

Cesar C. Palerm [*,1] **B. Wayne Bequette** [**]

[*] *School of Engineering, Rensselaer Polytechnic Institute*
[**] *Howard P. Isermann Department of Chemical Engineering, Rensselaer Polytechnic Institute*

Abstract: Direct model reference adaptive control (DMRAC) is an attractive algorithm that uses a linear combination of feedforward model states and command inputs, as well as the error between plant and model outputs; it does not require full state access nor observers. Real world applications are usually subject to limits on the controlled variables for either safety or practical reasons. Because of the unique structure of the controller, many of the methods developed to deal with saturation constraints can not be applied directly to DMRAC. This papers presents extensions to the algorithm to allow for the explicit handling of such constraints.

Keywords: Direct, Model Reference Adaptive Control, Saturation, Limits

1. INTRODUCTION

The controller is based on a simple adaptive control approach of MIMO plants first proposed by Sobel, Kaufman, and Mabius (Sobel *et al.*, 1979). This control structure uses a linear combination of feedforward model states and command inputs and feedback of the error between plant and model outputs. This class of algorithms requires neither full state access nor adaptive observers. Other important properties of this class of algorithms include (1) their applicability to non-minimum phase systems and (2) the fact that the plant (physical system) order may be much higher than the order of the reference model. Its ease of implementation and inherent robustness properties make this simple adaptive control approach attractive.

One of the main drawbacks of the standard DMRAC algorithm is its inability to handle input constraints. A related problem is tackled by (Bodson and Pohlchuck, 1998) for *indirect* model reference adaptive control. They propose four methods to deal with rate and saturation limits of aircraft actuators. The first one is simply to scale the control inputs while preserving the directionality of the commands. This is not a new idea — see, for example, (Åström and Rundqwist, 1989) — and it specially makes sense for systems with tight coupling. The second method is to relax the control requirements when the constraints are violated. Their control structure embeds the control requirements into a single constant k in a way that the larger its value the closed–loop poles are placed farther into the left–half plane. The third approach they use is to scale the reference inputs, once more maintaining the directionality of the control signal. The fourth method consists in approximating the accelerations that would be produced by the desired control inputs.

Of these, only the first approach can be applied to DMRAC, and even then it does not solve the windup problem. The other methods require an on–line estimate of the plant, which is not available in the DMRAC structure. In any case, all their methods highlight the sources of windup due to command limiting for all controllers. If

[1] Partially supported by the Consejo Nacional de Ciencia y Tecnología (CONACYT) of Mexico

the controller is too aggressive chances are that the control signal will saturate at some point; particularly if rate constraints are also imposed.

Based on the above ideas, the following are some specific ways in which command limiting can be dealt with in the DMRAC structure. Scaling back the control signal while keeping the directionality is a good start, but windup of the adaptive gains has to be handled. If the signal is just clipped, this will mean that the system outputs will not change as fast as the controller is expecting them to do, and the adaptation mechanism will keep increasing the gains.

This papers presents two specific methods to deal with saturation constraints. Formulation of the DMRAC algorithm is discussed in section 2, the extensions are presented in section 3, simulation results for some standard case problems are presented in section 4. Finally, results are discussed, and conclusions are drawn in section 5.

2. FORMULATION OF THE DMRAC ALGORITHM

The linear time invariant model reference adaptive control problem is considered for the plant

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \end{aligned} \tag{1}$$

where $x(t)$ is the $(n \times 1)$ state vector, $u(t)$ is the $(m \times 1)$ control vector, $y(t)$ is the $(q \times 1)$ plant output vector, and A, B are matrices with appropriate dimensions. The range of the plant parameters is assumed to be known and bounded by

$$\begin{aligned} &\underline{a}_{ij} \leq a(i,j) \leq \overline{a}_{ij} \qquad i,j = 1,\dots,n \\ &\underline{b}_{ij} \leq b(i,j) \leq \overline{b}_{ij} \qquad i = 1,\dots,n; j = 1,\dots,m \end{aligned} \tag{2}$$

The objective is to find, without explicit knowledge of A and B, the control $u(t)$ such that the plant output vector $y(t)$ follows the reference model

$$\begin{aligned} \dot{x}_m(t) &= A_m x_m(t) + B_m r(t) \\ y_m(t) &= C_m x_m(t) \end{aligned} \tag{3}$$

The model incorporates the desired behavior of the plant, but its choice is not restricted. In particular, the order of the plant may be much larger than the order of the reference model.

The adaptive control algorithm presented is based on the command generator tracker (CGT) concept developed by (Broussard and O'Brien, 1979). In the CGT method, it is assumed that there exists an ideal plant with ideal state and control trajectories, $x^*(t)$ and $u^*(t)$, respectively, which corresponds to perfect output tracking (i.e. when $y(t) = y_m(t) \ \forall \ t \geq 0$). By definition, this ideal plant satisfies the same dynamics as the real plant, and the ideal plant output is identically equal to the model output. Thus,

$$\dot{x}^*(t) = Ax^*(t) + Bu^*(t) \quad \forall \quad t \geq 0 \tag{4}$$

and

$$y^*(t) = y_m(t) = Cx^*(t) = C_m x_m(t) \tag{5}$$

Hence, when perfect tracking occurs, the real plant trajectories become the ideal plant trajectories, and the real plant output becomes the ideal plant output, which is defined to be the model output.

The ideal control law $u^*(t)$, generating perfect output tracking and the ideal state trajectories $x^*(t)$ is assumed to be a linear combination of the model states and model input:

$$\begin{bmatrix} x^*(t) \\ u^*(t) \end{bmatrix} = \begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix} \begin{bmatrix} x_m(t) \\ r(t) \end{bmatrix} \tag{6}$$

where the S_{ij} submatrices satisfy the following conditions

$$\begin{aligned} S_{11}A_m &= AS_{11} + BS_{21} \\ S_{11}B_m &= AS_{12} + BS_{22} \\ C_m &= CS_{11} \\ 0 &= CS_{12} \end{aligned} \tag{7}$$

In summary, when perfect output tracking occurs, $x(t) = x^*(t)$, and the ideal control is given by

$$u^*(t) = S_{21}x_m(t) + S_{22}r(t) \tag{8}$$

If when perfect output tracking does not occur, $y(t) \neq y_m(t)$, asymptotic tracking is achievable provided stabilizing output feedback is included in the control law

$$u(t) = S_{21}x_m(t) + S_{22}r(t) + K_e(y_m(t) - y(t)) \tag{9}$$

Then the adaptive control law based on this command generator tracker (CGT) approach is given as (Kaufman *et al.*, 1998)

$$u(t) = K_e(t)[y_m(t) - y(t)] + K_x(t)x_m(t) + K_r(t)r(t) \tag{10}$$

where $K_e(t)$, $K_x(t)$, and $K_r(t)$ are adaptive gains and concatenated into the matrix $K(t)$ as follows

$$K(t) = [K_e(t) \quad K_x(t) \quad K_r(t)] \tag{11}$$

Defining the vector $v(t)$ as

$$v(t) = \begin{bmatrix} y_m(t) - y(t) \\ x_m(t) \\ r(t) \end{bmatrix} \tag{12}$$

the control $u(t)$ is written in a compact form as follows

$$u(t) = K(t)v(t) \tag{13}$$

The adaptive gains are obtained as a combination of an integral gain and a proportional gain as shown below (Kaufman *et al.*, 1998)

$$K(t) = K_P(t) + K_I(t) \tag{14a}$$

$$K_P(t) = [y_m(t) - y(t)]v^T(t)\overline{T}, \quad \overline{T} \geq 0 \tag{14b}$$

$$\dot{K}_I(t) = [y_m(t) - y(t)]v^T(t)T, \quad T > 0 \tag{14c}$$

Where $\overline{T}$ and T are time invariant weighting matrices.

The sufficiency conditions for asymptotic tracking are

(1) There exists a solution to the CGT problem (eq. 7)
(2) The plant is ASPR; this is, there exists a positive definite constant gain matrix K_E, not needed for implementation, such that the closed loop transfer function

$$G(s) = [I + G_p(s)K_E]^{-1} G_p(s) \tag{15}$$

is strictly positive real (SPR).

In general, the ASPR condition is not satisfied by most real systems. (Bar-Kana and Kaufman, 1985) have shown that a non–ASPR plant of the form $G_p(s) = C(sI - A)^{-1}B$ can be augmented with a feedforward compensator $H(s)$ such that the augmented plant transfer function

$$G_a(s) = G_p(s) + H(s) \tag{16}$$

is ASPR. However the resulting adaptive controller will in general result in a model following error that is bounded but not zero in steady state. To eliminate this problem, a modification that incorporates the supplementary feedforward into the reference model output as well as the plant output has been developed by (Kaufman and Neat, 1993).

3. EXTENSIONS TO HANDLE SATURATION CONSTRAINTS

3.1 *Saturation Limiting Method 1*

The first proposed method for dealing with saturation limits is the easiest to implement. Whenever the command saturates, the gains keep changing according to the dynamics of (14). If $y_m(t) - y(t) \neq 0$, then the gains will keep increasing or decreasing, even though these changes are not having any effect on the system's output. Thus, the most straightforward way of dealing with this is to stop the adaptation process whenever the control signal saturates. Modifying the adaptation law as follows

$$K(t) = K_P(t) + K_I(t) \tag{17a}$$

$$K_P(t) = [y_m(t) - y(t)]v^T(t)\overline{T}\kappa, \quad \overline{T} \geq 0 \tag{17b}$$

$$\dot{K}_I(t) = [y_m(t) - y(t)]v^T(t)T\kappa, \quad T > 0 \tag{17c}$$

where κ is defined as

$$\kappa = \begin{cases} 1 & \text{if } u(t) \text{ does not saturate,} \\ 0 & \text{if } u(t) \text{ saturates} \end{cases} \tag{18}$$

In the case of MIMO systems the control signal $u(t)$ should be considered to saturate if any of its elements saturates. In this case all of the inputs should be scaled back to maintain the directionality of the the command.

3.2 *Saturation Limiting Method 2*

The second method maintains the adaptation mechanism alive, but bleeds the integral part (14c) to minimize (or eliminate) the windup. In this case the adaptation law is changed to

$$K(t) = K_P(t) + K_I(t) \tag{19a}$$

$$K_P(t) = [y_m(t) - y(t)]v^T(t)\overline{T}, \quad \overline{T} \geq 0 \tag{19b}$$

$$\dot{K}_I(t) = [y_m(t) - y(t)]v^T(t)T - \sigma(t)K_I(t), \quad T > 0 \tag{19c}$$

The $\sigma(t)$ term is added to bleed the gains when the saturation constraints are hit. A basic form for this function is

$$\dot{\sigma}(t) = K_{\sigma_1} |u(t) - u_{sat}(u(t))| - K_{\sigma_2}\sigma(t) \tag{20}$$

with

$$u_{sat}(u(t)) = \begin{cases} u_{lb} & \text{for } u(t) < u_{lb} \\ u(t) & \text{for } u_{lb} \leq u(t) \leq u_{ub} \\ u_{ub} & \text{for } u(t) > u_{ub} \end{cases} \tag{21}$$

where $K_{\sigma_1} > 0$ and $K_{\sigma_2} > 0$ are tuning parameters, and u_{lb} and u_{ub} are the lower and upper saturation limits respectively. Thus, when the command is within the saturation limits $\sigma(t) \to 0$ and increases when the they are violated.

For the MIMO case $\sigma(t)$ can be defined as a diagonal matrix with elements

$$\dot{\sigma}_{ii}(t) = K_{\sigma_1} |u_i(t) - u_{i_{sat}}(u_i(t))| - K_{\sigma_2}\sigma_{ii}(t) \tag{22}$$

An obvious limitation to this approach is the addition of two additional tuning parameters (which are matrices in the MIMO case).

3.3 *Preserving Directionality*

As noted before, preserving the directionality of the control signal in MIMO systems is important, particularly when the system is tightly coupled. This is quite simple to accomplish, and just consists of scaling all control inputs proportionally, using the most severely saturated signal as the base.

For each control signal we can find a constant $\rho, \quad 0 \leq \rho \leq 1$ such that

$$u_{i_{sat}} = \rho_i u_i \tag{23}$$

then scale the control input vector using

$$u_{sat} = min\,(\rho_i)\, u \tag{24}$$

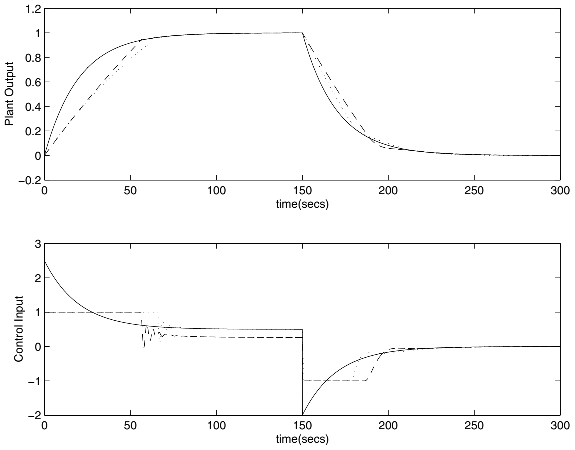

Fig. 1. SISO Case: solid is unconstrained, dashed uses method 1 and dotted uses method 2

4. SIMULATION RESULTS

For the simulations simple systems are used for illustrative purpose, avoiding the need for the feedforward compensators. These examples are taken from the literature of anti–windup compensation, namely from (Zheng *et al.*, 1994).

4.1 *Example 1 - SISO Case*

Consider the plant given by

$$\begin{aligned} \dot{x}(t) &= -0.01x(t) + 0.125u(t) \\ y(t) &= 0.16x(t) \end{aligned} \tag{25}$$

and the plant is to follow the output of the reference model

$$\begin{aligned} \dot{x}_m(t) &= -0.05x_m(t) + 0.25r(t) \\ y_m(t) &= 0.2x_m(t) \end{aligned} \tag{26}$$

Using the solution to the CGT problem (eq. 7), the initial condition for the gain is

$$K(0) = [5 \quad -0.4 \quad 2.5]$$

and the adaptation gains are set to

$$\begin{aligned} \overline{T} &= [0.5 \quad 0.5 \quad 0.5] \\ T &= [50 \quad 5 \quad 50] \end{aligned}$$

For method 2 the additional parameters are selected as

$$K_{\sigma_1} = 0.01 \qquad K_{\sigma_2} = 0.1$$

The input is constrained to be $u \in [-1, 1]$. The setpoint is 1 at $t = 0$ sec and is then set to 0 at $t = 150$ sec. Figure 1 shows the simulation results for the cases when the system is unconstrained (solid), constrained using method 1 (dashed) and constrained using method 2 (dotted). Both methods give comparable results; if anything method 2 does a slightly better job, but at the expense of two additional tuning parameters.

4.2 *Example 2 - MIMO Case*

Consider the MIMO plant given by

$$\begin{aligned} \dot{x}(t) &= \begin{bmatrix} -0.01 & 0 \\ 0 & -0.01 \end{bmatrix} x(t) + \begin{bmatrix} 0.5 & 0 \\ 0 & 1 \end{bmatrix} u(t) \\ y(t) &= \begin{bmatrix} 0.8 & -0.5 \\ -0.6 & 0.4 \end{bmatrix} x(t) \end{aligned} \tag{27}$$

and the plant is to follow the output of the reference model

$$\begin{aligned} \dot{x}_m(t) &= \begin{bmatrix} -0.05 & 0 \\ 0 & -0.05 \end{bmatrix} x_m(t) + \begin{bmatrix} 0.25 & 0 \\ 0 & 0.25 \end{bmatrix} r(t) \\ y_m(t) &= \begin{bmatrix} 0.2 & 0 \\ 0 & 0.2 \end{bmatrix} x_m(t) \end{aligned} \tag{28}$$

Using the solution to the CGT problem (eq. 7), the initial condition for the gain is

$$K(0) = \begin{bmatrix} 0.2 & 0.1 & 2 & 2.5 & -0.32 & -0.4 \\ 0.2 & 0.2 & 1.5 & 2 & -0.24 & -0.32 \end{bmatrix}$$

and the adaptation gains are set to

$$\begin{aligned} \overline{T} &= \begin{bmatrix} 5\text{E-}4 & 0 & 2.5\text{E-}2 & 0 & 5\text{E-}4 & 0 \\ 0 & 5\text{E-}4 & 0 & 2.5\text{E-}2 & 0 & 5\text{E-}4 \end{bmatrix} \\ T &= \begin{bmatrix} 5\text{E-}3 & 0 & 5\text{E-}4 & 0 & 5\text{E-}3 & 0 \\ 0 & 5\text{E-}3 & 0 & 5\text{E-}4 & 0 & 5\text{E-}3 \end{bmatrix} \end{aligned}$$

For method 2 the additional parameters are selected as

$$\begin{aligned} K_{\sigma_1} &= \begin{bmatrix} 9\text{E-}5 & 0 \\ 0 & 9\text{E-}5 \end{bmatrix} \\ K_{\sigma_2} &= \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix} \end{aligned}$$

The input is constrained to be $u_i \in [-1, 1]$.

For method 1 the simulation results are shown in figure 2; it shows the response of the ucosntrained system (solid), the response when directionality is preserved (dashed) and when it is not (dotted). The setpoint is set to $[0.63 \quad 0.79]^T$ at $t = 0$ sec and is then set back to $[0 \quad 0]^T$ at $t = 200$ sec. The large overshoot for both outputs when the directionality is not preserved highlights the need to preserve it.

For method 2 the simulation results are shown in figure 3; it shows the response of the ucosntrained system (solid), the response when directionality is preserved (dashed) and when it is not (dotted). The setpoint is set to $[0.63 \quad 0.79]^T$ at $t = 0$ sec and is then set back to $[0 \quad 0]^T$ at $t = 300$ sec. Once more, preserving the directionality of the control signal has a big impact on the system's response.

Focusing on the cases when directionality is preserved, the overshoots of the outputs using method 2 are significantly smaller than using method 1. At the same time, it takes method 2 a longer time to come to steady state; this is due to the integral factor in the compensation for the saturation of the signal. This could be adjusted by

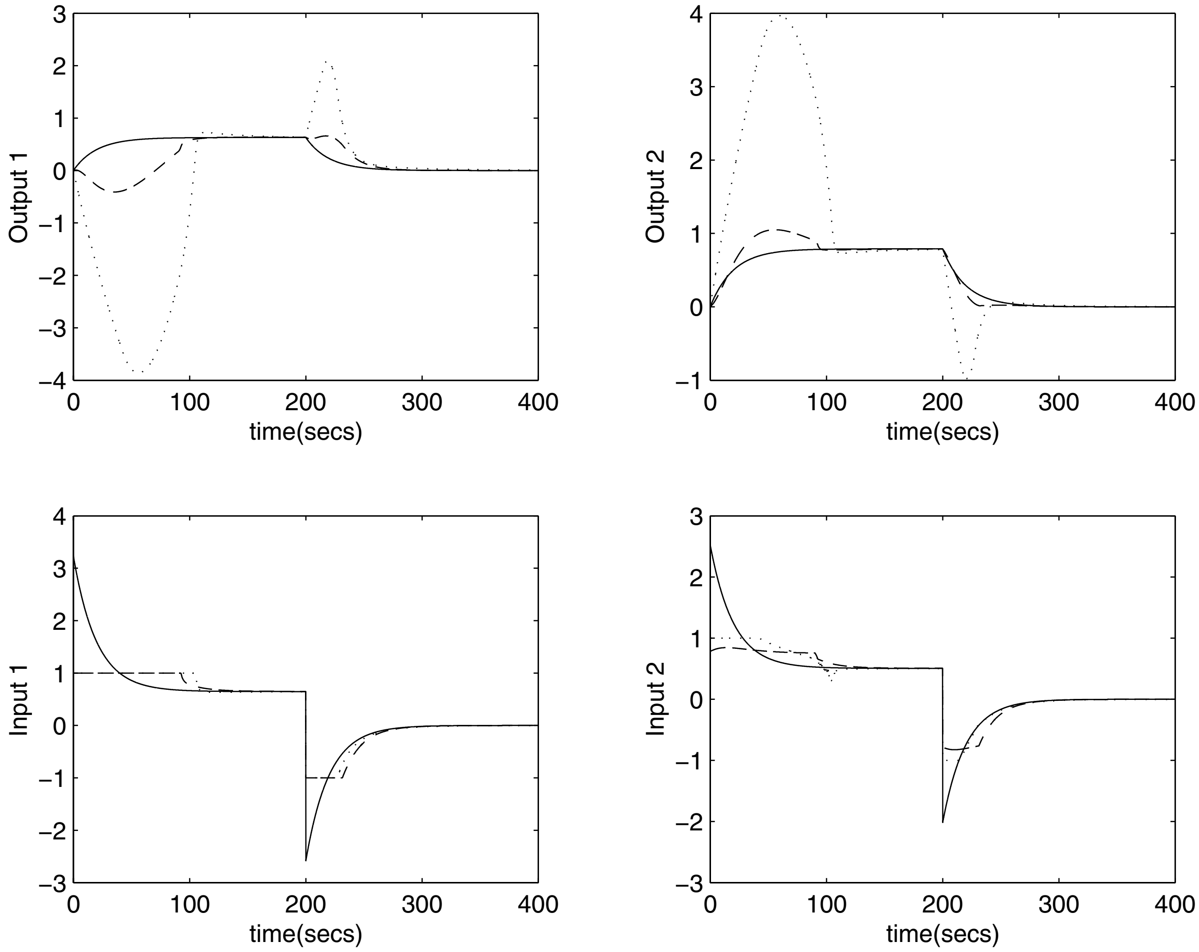

Fig. 2. MIMO Case using method 1: solid is unconstrained, dashed is constrained while preserving directionality, dotted is unconstrained without directionality

tuning K_{σ_2}, but at the expense of the performance when the constraints are hit.

5. CONCLUSIONS

This paper presents two specific methods that extend the algorithm for Direct Model Reference Adaptive Control to systems that present saturation constraints on the command signal. Both methods give good results, with their own unique strengths and weaknesses. Method 1 is the simplest to implement, as adaptation is halted as long as the saturation constraint is hit and the command signals are clipped while preserving directionality. Method 2 tends to give better results, but at the expense of added parameters that need to be tuned. For systems with many inputs and outputs this can quickly become a nuisance to deal with.

There are still other avenues to be explored. Modifying the reference model to relax the requirements is one possibility. Work in progress looks at parameterizing the reference model in a way that the performance requirements can be relaxed; this has to be accomplished without access to on–line plant identification and preserving the stability of the closed–loop system.

The two methods presented above can also be applied when limits on the rate of change of the control signal are imposed. In this case method 2 is more effective, as it only modulates how fast the adaptation mechanism is going to adjust the controller gains. Simply halting the adaptation as in method 1 means that the controller is not adjusting according to the plant's exhibited dynamics. It could be that the actual operating condition of the plant is such that significant changes have to be made on the controller gains; halting the adaptation will certainly not accomplish this.

6. REFERENCES

Åström, Karl Johan and Lars Rundqwist (1989). Integrator windup and how to avoid it. In: *Proceedings of the American Control Conference*. Pittsburgh, PA. pp. 1693–1698.

Bar-Kana, Izhak and Howard Kaufman (1985). Global stability and performance of a simplified adaptive algorithm. *Int. J. Control* **42**(6), 1491–1505.

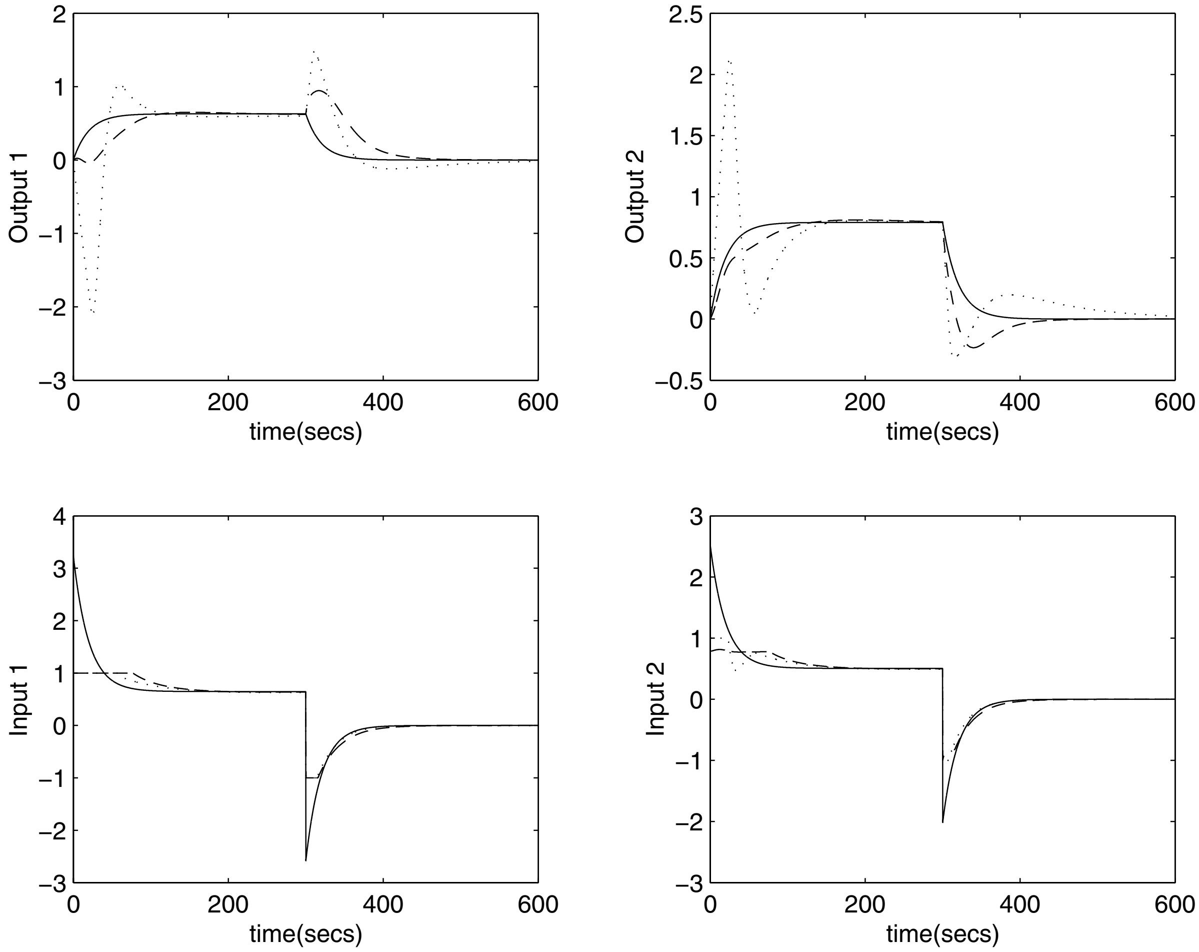

Fig. 3. MIMO Case using method 2: solid is unconstrained, dashed is constrained while preserving directionality, dotted is unconstrained without directionality

Bodson, Marc and William A. Pohlchuck (1998). Command limiting in reconfigurable flight control. *Journal of Guidance, Control, and Dynamics* **21**(4), 639–646.

Broussard, J. and O. O'Brien (1979). Feedforward control to track the output of a forced model. In: *17th IEEE Conference on Decision and Control*. San Diego, CA. pp. 1149–1155.

Kaufman, Howard and Gregory W. Neat (1993). Asymptotically stable multiple–input multiple–output direct model reference adaptive controller for processes not necessarily satisfying a positive real constraint. *Int. J. Control* **58**(5), 1011–1031.

Kaufman, Howard, Izhak Bar-Kana and K. Sobel (1998). *Direct Adaptive Control Algorithms*. 2nd ed.. Springer Verlag.

Sobel, K., Howard Kaufman and L. Mabius (1979). Model reference output adaptive control systems without parameter identification. In: *18th IEEE Conference on Decision and Control*. Ft. Lauderdale, FL. pp. 347–351.

Zheng, Alex, Mayuresh V. Kothare and Manfred Morari (1994). Anti–windup design for internal model control. *International Journal of Control* **60**(5), 1015–1024.