Bulanık Sistemler ve Kontrol (devamı)

Dayanıklı (Gürbüz) Kontrol

Demiryolu Kontrol Sistemleri

Denizcilik Navigasyon Sistemleri

Doğrusal Kontrol Sistemleri

Adaptive Robust Controller Design For Non-minimum Phase Systems

M. Ataollahi and H. D. Taghirad

***Abstract*— Based on the synthesis algorithm of dynamical backstepping design procedure, in this paper a new adaptive robust approach for non-minimum phase systems is proposed. The proposed controller consists of two parts; a backstepping controller as the robust part and a model reference (MRAS) controller as the adaptive part. In this control scheme the adaptive part acts not only as a medium to converge to suitable values for the unknown parameters and to reduce the uncertainty, but also provides a minimum-phase model for the robust controller to be well stabilized. A simulation case study is studied to show how to perform the proposed control law, and to illustrate the effectiveness of this method compared to that of conventional robust controllers.**

***Index Terms*—Adaptive robust controller, model reference adaptive systems (MRAS), backstepping, non-minimum phase systems.**

I. Introduction

DEMAND for high performance in systems with nonlinear behavior and model uncertainties is one of most challenging area in control theory. An adaptive robust controller (ARC) represents a systematic way to design a controller for such requirement, and it combines adaptive and robust control approaches to preserve the advantages of the both methods while overcoming their drawbacks [1]. Alternatives of ARC controllers have been developed in literature[2]. The saturated adaptive robust controllers (SARC) developed in [3] for uncertain nonlinear systems in the presence of practical constraint of control input saturation. Also the output feedback ARC schemes that need the output measurement sensor only are developed in [4].

Another approach that has been developed in [5], is to combine the ARC control with dynamic backstepping method. In this method the robust controller is used as the main controller for trajectory tracking, and adaptive controller tries to decrease the uncertainty and helps to reduce tracking error especially at steady state [6]. This method is used to control hard disk drives in [7]. Although this approach is very promising in practice, it suffers from a stringent limitation that cannot be applied to non-minimum phase systems.

In this paper, an ARC backstepping method is proposed to guarantee the stability of non-minimum phase systems. In order to accomplish this task a model reference adaptive systems (MRAS) in addition to a robust controller to reclaim the unstructured uncertainties and disturbances. Simulation study shows how to implement such controller, and furthermore, illustrates the effectiveness of the proposed controller in comparison to a conventional robust controller for a non-minimum phase system.

Authors are with the Advanced Robotics and Automated Systems (ARAS), Department of Systems and Control, Faculty of Electrical and Computer Engineering, K. N. Toosi University of Technology, email: taghirad@kntu.ac.ir.

II. Control Structure

A. Problem Statement

Consider a SISO system described by a nominal model and multiplicative uncertainty

$$y(t) = \frac{B(s)}{A(s)}u(t) + W(s)\Delta(y,t) \tag{1}$$

in which

$$A(s) = s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0 \tag{2}$$

$$B(s) = b_m s^m + b_{m-1}s^{m-1} + \cdots + b_1 s + b_0 \tag{3}$$

where $m < n$. The plant parameters a_i's and b_i's are unknown constants, d_y is the output disturbance and $\Delta(y,t)$ represents any disturbance coming from the intermediate channels of the plant. The state space representation of the plant (1) is given as follows:

$$\begin{aligned} \dot{x}_1 &= x_2 - a_{n-1}x_1 + \Delta_1 \\ &\vdots \\ \dot{x}_\rho &= x_{\rho+1} - a_m x_1 + \Delta_\rho + b_m u \\ &\vdots \\ \dot{x}_n &= -a_0 x_1 + \Delta_n + b_0 u \\ y &= x_1 + d_y \end{aligned} \tag{4}$$

Note that in this representation, the uncertainty profile $W(s)$ must be transformed to state space uncertainties $\Delta_i{}'$s. Let's define the vector of uncertainty as below

$$\Delta = [\Delta_1 \quad \cdots \quad \Delta_n]^T \tag{5}$$

The following standard assumptions indicate the framework of the system and nonlinearities in which the system is incorporating:

Plant is with order n, relative degree ρ, could be *non-minimum phase*, and the sign of b_m is known. The extent of uncertain nonlinearities, Δ, d_y and $\dot{d}_y$, are known, i.e.,

$$\begin{aligned} \Delta &\in \Omega_\Delta \triangleq \{\Delta \mid \|\Delta\| < \delta(t)\} \\ d_y &\in \Omega_d \triangleq \{d_y \mid \|d_y\| < \delta_d(t)\} \\ \dot{d}_y &\in \Omega_f \triangleq \{\dot{d}_y \mid \|\dot{d}_y\| < \delta_f(t)\} \end{aligned} \tag{6}$$

Where, δ, δ_d, δ_f are assumed to be known. Given the reference trajectory, $y_r(t)$, the objective of the controller design is to synthesize a control signal, $u(t)$, such that output $y(t)$ tracks the reference trajectory as closely as possible, in spite of various model uncertainties. The reference trajectory and its derivatives up to n are assumed to be known, bounded, and piecewise continuous.

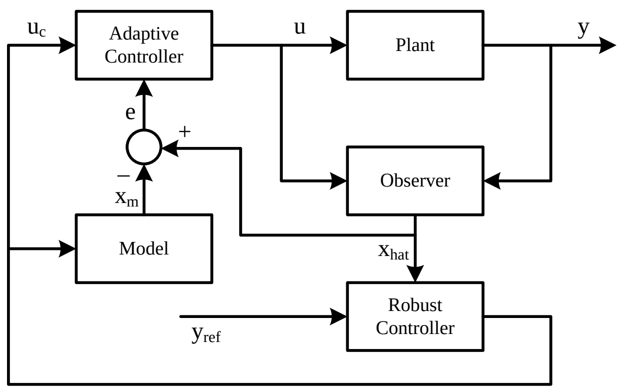

Fig. 1. The structure of ARC controller using MRAS and backstepping method. Adaptive controller forces the plant to track the reference model, and robust controller stabilizes the perturbed plant.

III. Adaptive Part

Since only the output $y(t)$, is measured and since , the full state information of the system is required, a *Kreisselmeier* observer [8] may be used to observe the states from the outputs. This approach proceeds from a so-called parameterized observer, which is only an alternative to the customary representation of the *Luenberger* observer. Note that any observer that can estimate states of a perturbed system can be used as other alternatives.

A. Controller Design

In this section we describe a systematic algorithm to design an ARC output tracking controller that consists of two parts:

1) Adaptive part: This part is designed by a model reference adaptive system (MRAS) controller which tackles the parametric uncertainty.
2) Robust part: This part may be designed from the rich theory of robust control, to compensate unstructured uncertainty, disturbances, state estimation errors and tracking error of MRAS controller. In here a backstepping controller is proposed.

The block diagram of the ARC controller is shown in Fig. 1.

B. MRAS Design for State Space Models

Model reference adaptive system is one of the most celebrated adaptive controllers. In this method the required performance is defined with respect to a reference model, and the controller forces the plant to behave as the reference model. In this paper a state space representation of MRAS is used, in which the system states have to track the reference model states. A general scheme of this method is shown in Fig. 2. The closed loop system consists of two loops: An ordinary feedback loop consisting of plant and controller, and the adaptation loop that suitably changes the controller parameters. Adaptation mechanism compares plant states and model states, and updates the controller parameters to reduce the tracking error of the states. The adaptation rules are obtained using *Lyapunov* stability theorem. Here this method is briefly reviewed from [9].

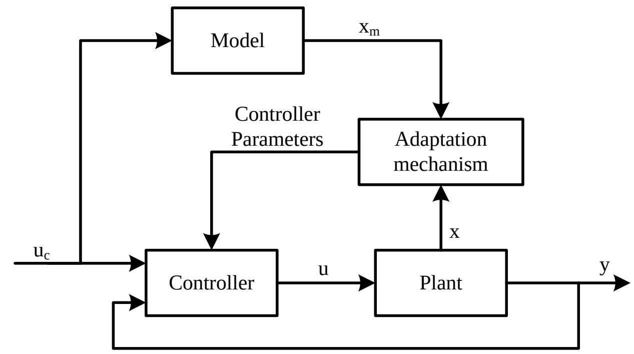

Fig. 2. A model reference adaptive system represented in state space. The adaptation mechanism compares system states to model states and updates controller parameters.

Consider the linear SISO system described by

$$\dot{x} = Ax + Bu \tag{7}$$

whose states should track the states of

$$\dot{x}_m = A_m x_m + B_m u_c \tag{8}$$

using the control law

$$u = M u_c - Lx \tag{9}$$

in which, M is a pre-compensator to eliminate steady state error and L is gain of state feedback. The closed loop system will be

$$\dot{x} = (A - BL)x + BMu_c = A_c(\theta)x + B_c(\theta)u_c \tag{10}$$

in which the vector θ contains controller parameters M and L. Compare (8) and (10) to find an appropriate value for θ. A sufficient condition to have a suitable value for θ is to find θ^0 that satisfies

$$A_c(\theta^0) = A_m \Longrightarrow\ A - BL^0 = A_m \tag{11}$$

$$B_c(\theta^0) = B_m \Longrightarrow\ BM^0 = B_m \tag{12}$$

These conditions are called compatibility conditions. The reference model must be chosen such that we can find M^0 and L^0 for initial condition.

C. Error Equation Formation and Adaptation Rule

To design a model reference controller, first define the error equation

$$e = x - x_m \tag{13}$$

Now write the derivative of error as below,

$$\begin{aligned}\dot{e} = \dot{x} - \dot{x}_m &= Ax + Bu - A_m x_m - B_m u_c \pm A_m x \\ &= A_m e + (A - A_m - BL)x + (BM - B_m)u_c \\ &= A_m e + (A_c(\theta) - A_m)x \\ &\quad + (B_c(\theta) - B_m)u_c \\ &= A_m e + \Psi(\theta - \theta^0)\end{aligned} \tag{14}$$

in which, the matrix Ψ contains generally states, output, reference input and their derivatives.

Now consider the following Lyapunov function for the system

$$V(e,\theta) = \frac{1}{2}\left(\gamma e^T P e + (\theta - \theta^0)^T(\theta - \theta^0)\right) \tag{15}$$

in which, P is a positive definite matrix. The derivative of this function is

$$\dot{V} = -\frac{1}{2}\gamma e^T Q e + \gamma(\theta - \theta^0)^T \Psi^T P e + (\theta - \theta^0)^T \dot{\theta} \quad (16)$$
$$= -\frac{1}{2}\gamma e^T Q e + (\theta - \theta^0)^T \left(\dot{\theta} + \gamma \Psi^{\mathrm{T}} P e\right)$$

in which Q is a positive definite matrix that satisfies:

$$A_m^T P + P A_m = -Q \quad (17)$$

Note that if A_m is *Hurwitz*, a pair of P and Q exists to satisfy the equation above. Now choose the following adaptation rule

$$\dot{\theta} = -\gamma \Psi^{\mathrm{T}} P e \quad (18)$$

This immediately leads to

$$\dot{V} = -\frac{1}{2}\gamma e^T Q e \quad (19)$$

which is negative definite and makes the closed loop system stable. In [9] by using Barbalat's Lemma, it is shown that the tracking error goes to zero. In (18) the parameter γ is a designing parameter that tunes the adaptation speed. This parameter is positive and not too small; otherwise, the adaptation does not work well. On the other hand if it is set too large, the adaptation could not achieve the parameter convergence properly and it leads to oscillate the system output and even destabilize the system. The matrix Ψ consists of two: One part is for state feedback and another is for pre-compensator.

$$\Psi(\theta - \theta^0) = (A - A_m - BL)x + (BM - B_m)u_c \quad (20)$$

Simplify each part separately, for the first part we have

$$(A - A_m - BL)x = (BL^0 - BL)x = -B(L - L^0)x. \quad (21)$$

Next we define $L - L^0$ as below

$$L - L^0 = [l_1 \quad l_2 \quad \dots \quad l_n] \quad (22)$$

Substitute $L - L^0$ in (21) :

$$-B(L - L^0)x = -B[l_1 \quad l_2 \quad \dots \quad l_n]\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad (23)$$
$$= -Bx^T(L - L^0)^T$$

For the second part of (20), because the system is SISO, the matrix M is scalar, so we can write

$$(BM - B_m)u_c = (BM - BM^0)u_c = Bu_c(M - M^0) \quad (24)$$

Using (23) and (24), we can rewrite (20) as follows

$$\Psi(\theta - \theta^0) = (A - A_m - BL)x + (BM - B_m)u_c$$
$$= [-Bx^T \quad Bu_c]\begin{bmatrix} (L - L^0)^T \\ M - M^0 \end{bmatrix} \quad (25)$$

Define the parameter vector as

$$\theta \triangleq \begin{bmatrix} L^T \\ \mathrm{M} \end{bmatrix} \quad (26)$$

Then,

$$\Psi = B[-x^T \quad u_c] \quad (27)$$

Now complete the adaptation rule expressed in (18) using Ψ.

In this method a necessary condition for stability is that the sign of b_0 in the plant and the reference model must be equal [9]. Also, the reference model must be stable [9] and minimum phase, since we will use this reference model in robust controller design.

IV. Robust Part

As said before this part can be any robust controller, but a backstepping controller is proposed in this paper. The general idea of such controller is developed in [5], however, for non-minimum phase systems, the controller should be assigned for the reference model. This means that all the parameters and states that is used in this framework, corresponds to that of the reference model, and not the original model.

The backstepping procedure is an iterative method. Introducing the positive constants $c_i, i = 1, \dots, n$ as design parameters, we can follow the following design procedure.

***Step 1*:** The design procedure takes advantages of improved backstepping via observer design in [10], and dynamic backstepping in [5]. This procedure starts with the system dynamics, and we derive the differentiation of the output tracking error

$$z_1 = y_m(t) - y_r(t) \quad (28)$$

by

$$\dot{z}_1 = x_2 - a_{n-1}x_1 + \Delta_1 + \dot{d}_y - \dot{y}_r(t) \quad (29)$$

In this equation the control input variable does not show up yet, and it cannot be directly stabilized. Hence, Choose one of the parameters appearing in the equation to treat as the virtual input as in usual backstepping procedures. One choice can be x_2, since dynamic equations of (4) shows that the actual control u appears only after $n - m$ differentiation of it, this choice appears earlier than any other parameter in this equation. Assuming x_2 as the virtual control input, a control law α_1 can be designed to stabilize equation (29). As x_2 is not the actual control, we define z_2 as the error between actual and desired value of it

$$z_2 = \hat{x}_2 - \alpha_1 \quad (30)$$

Now we can synthesize a virtual control law

$$\alpha_1 = \alpha_{1a} + \alpha_{1s} \quad (31)$$

to force z_1 to become small in spite of the various system uncertainties. For this reason, we separate the virtual control input α_1 into two parts, in which the first term is designed to deal with the error dynamics, and the second term guarantees the robust stability of the system in presence of uncertain dynamics. Generally, define

$$\tilde{x}_i = x_i - \hat{x}_i \quad (32)$$

This error consists of two components, the estimation error of the observer and the tracking error of the reference model states. We use the reference model to design this controller, but finally we employ the real states to the tracking error of adaptive controller. Therefore, rewrite (29) as

$$\dot{z}_1 = z_2 + \alpha_{1a} + \alpha_{1s} + \tilde{x}_2 - a_{n-1}\hat{x}_1 - a_{n-1}\tilde{x}_1 + \Delta_1 + \dot{d}_y - \dot{y}_r(t) \tag{33}$$

and set

$$\alpha_{1a} = a_{n-1}\hat{x}_1 + \dot{y}_r(t) - c_1 z_1 \tag{34}$$

This is the first subsystem to be stabilized using the virtual input α_1. In order to do this, the following Lyapunov function is proposed.

$$V_1 = \frac{1}{2} z_1^2 \tag{35}$$

Differentiate the Lyapunov function as

$$\dot{V}_1 = z_1\big(z_2 - c_1 z_1 + \alpha_{1s} + \tilde{x}_2 - a_{n-1}\tilde{x}_1 + \Delta_1 + \dot{d}_y\big) \tag{36}$$

Therefore,

$$\dot{V}_1 = z_1 z_2 - c_1 z_1^2 + z_1\big(\alpha_{1s} + \tilde{x}_2 - a_{n-1}\tilde{x}_1 + \Delta_1 + \dot{d}_y\big) \tag{37}$$

Since $\|\Delta_1\| \le \delta_1(t)$, $\|d_y\| < \delta_d(t)$ and $\|\dot{d}_y\| < \delta_f(t)$ are known, there exist a robust control function α_{1s}, satisfying the following conditions:

$$z_1\big(\alpha_{1s} + \tilde{x}_2 - a_{n-1}\tilde{x}_1 + \Delta_1 + \dot{d}_y\big) < r_1 \tag{38}$$

$$\alpha_{1s} z_1 \le 0 \tag{39}$$

The parameter r_1 - and also, in the rest of paper r_i's - is an arbitrary design parameter indicating the boundary layer width of the sliding surface, and can be chosen arbitrarily small. Essentially, condition (38) shows that robust control input is synthesized to dominate the model uncertainties coming from Δ_1 with the level of control accuracy being measured by designating parameter r_1, and condition (39) ensures that α_{1s} is dissipating in nature so that it does not interfere with functionality of the adaptive control part α_{1a} [1], Examples of smooth α_{1s} satisfying (38) and (39) can be found in [2] and [11].

***Remark 1*:** One example of a smooth α_{1s} can be generated in the following way. Let $h_1(x,t)$ be any smooth function satisfying

$$h_1(x,t) \ge \|\tilde{x}_2\| + \|a_{n-1}\tilde{x}_1\| + \|\Delta_1\| + \|\dot{d}_y\| \tag{40}$$

Now it can be shown that

$$\alpha_{1s} = -h_1(x,t)\tanh\left(\frac{0.2785 h_1 . z_1}{r_1}\right) \tag{41}$$

satisfies this condition. The following steps of the backstepping procedure also requires the introduction of a robustly stabilizing control term α_{is} which also uses the relevant parameter r_i in relation to the system dynamical functions, states, Δ_i, and z_i.

***Step 2*:** Develop the equation of second error dynamics as:

$$\dot{z}_2 = \dot{x}_2 - \dot{\alpha}_1 \tag{42}$$

Since α_1 is measurable, therefore, $\dot{\alpha}_1$ is available. Hence, the second error subsystem can be rewritten as follows

$$\dot{z}_2 = x_3 - a_{n-2}x_1 + \Delta_2 - \dot{\alpha}_1 \tag{43}$$

Now the second Lyapunov function can be introduced as

$$V_2 = V_1 + \frac{1}{2} z_2^2 \tag{44}$$

Like the first step, stabilizing the system through Lyapunov function V_2, chose x_3 as the virtual input and introduce a new variable as the deviation of it with the desired control α_2. This virtual control also consists of two parts

$$\alpha_2 = \alpha_{2a} + \alpha_{2s} \tag{45}$$

With respect to deviation of virtual control to x_3

$$z_3 = \hat{x}_3 - \alpha_2 \tag{46}$$

Choose α_{2a} as

$$\alpha_{2a} \triangleq a_{n-2}\hat{x}_1 + \dot{\alpha}_1 - c_2 z_2 - z_1 \tag{47}$$

where, c_2 is positive constant gain. Substitute (47) into (43), and write the derivative of V_2 as

$$\dot{V}_2 = z_2 z_3 - c_1 z_1^2 - c_2 z_2^2 + z_1\big(\alpha_{1s} + \tilde{x}_2 - a_{n-1}\tilde{x}_1 + \Delta_1 + \dot{d}_y\big) + z_2(\alpha_{2s} + \tilde{x}_3 - a_{n-2}\tilde{x}_1 + \Delta_2) \tag{48}$$

Similar to step 1, we consider same conditions on the second part of control to meet the robust stability requirements.

***Step k*:** $(3 \le k \le \rho - 1)$**:** Mathematical induction can be used to prove the general results for all intermediate steps up to $\rho - 1$. At *Step* k , the same design as in the above two steps will be employed to construct the control function α_k for x_{k+1}. For these steps express derivation of

$$z_k = \hat{x}_k - \alpha_{k-1} \tag{49}$$

as

$$\dot{z}_k = x_{k+1} - a_{n-k}x_1 + \Delta_k - \dot{\alpha}_{k-1} \tag{50}$$

Treating x_{k+1} as the virtual control input, the compensation part α_{ka} is synthesized as in (47).

$$\alpha_{ka} \triangleq a_{n-k}\hat{x}_1 + \dot{\alpha}_{k-1} - c_k z_k - z_{k-1} \tag{51}$$

Using mathematical induction, the control input and time derivative of the Lyapunov function for step k can be considered similarly, and the k-th Lyapunov function may be defined as

$$V_k = V_{(k-1)} + \frac{1}{2} z_k^2 \tag{52}$$

The time derivative of this Lyapunov function may be written as

$$\dot{V}_k = z_k z_{(k+1)} - \sum_{i=1}^{k} c_i z_i^2 + z_1\big(\alpha_{1s} + \tilde{x}_2 - a_{n-1}\tilde{x}_1 + \Delta_1 + \dot{d}_y\big) + \sum_{i=2}^{k} z_i(\alpha_{is} + \tilde{x}_{i+1} - a_{n-i}\tilde{x}_1 + \Delta_i) \tag{53}$$

The robust control part, α_{ks}, is also chosen to satisfy conditions (38), in order to overcome the various system dynamical uncertainties or uncertain nonlinearities. This induction can be proven in the same way as in [1] and [8].

***Step* ρ:** This step is special, because in this step the actual control input u, appears for the first time in the backstepping design procedure. Like before define a deviation variable

$$z_\rho = \hat{x}_\rho - \alpha_{\rho-1} \tag{54}$$

Taking the known and unknown parts apart from $\dot{\alpha}_{\rho-1}$, the derivation of error parameter z_ρ can be written as

$$\dot{z}_\rho = x_{\rho+1} - a_m x_1 + \Delta_\rho + b_m u - \dot{\alpha}_{\rho-1} \tag{55}$$

In traditional backstepping algorithm this actual control input is used to stabilize the system and the design procedure is completed, however, similar to what is done in [5], continue the procedure. Suppose that $x_{\rho+1}$ is the virtual control input as before, we can define

$$\alpha_\rho = \alpha_{\rho a} + \alpha_{\rho s} \tag{56}$$

Moreover, $\alpha_{\rho a}$ is synthesized in the same way as in (51), except that it is extended by u:

$$\alpha_{\rho a} \triangleq a_m \hat{x}_1 - b_m u + \dot{\alpha}_{\rho-1} - c_\rho z_\rho - z_{\rho-1} \tag{57}$$

Using this control input, the derivative of this Lyapunov function will be the same as in (44) up to the very last step before, and similarly $\alpha_{\rho s}$ must satisfy the same conditions.

***Step* +j** $(\mathrm{j} \le \mathrm{m}-1)$**:** Continue the procedure like previous step, the control input itself and the derivatives of it will appear in the virtual control design at these steps. This will lead to imposing dynamics into the control input. Defining $z_{\rho+j}$ as the deviation between the virtual and proposed control input leads to

$$\dot{z}_{\rho+j} = x_{\rho+j+1} - a_{m-j} x_1 + \Delta_{\rho+j} + b_{m-j} u - \dot{\alpha}_{\rho+j-1} \tag{58}$$

Once more, the $\alpha_{(\rho+j)a}$ is synthesized in the same way as in (57):

$$\begin{aligned}\alpha_{(\rho+j)a} \triangleq\ & a_{m-j}\hat{x}_1 - b_{m-j}u + \dot{\alpha}_{\rho+j-1} - c_{\rho+j}z_{\rho+j} \\ & - z_{\rho+j-1}\end{aligned} \tag{59}$$

Step n: This is the final step of the design in which the dynamic output tracking control law will be synthesized. As the previous steps we express the derivative of z_n as

$$\dot{z}_n = -a_0 x_1 + \Delta_n + b_0 u - \dot{\alpha}_{n-1} \tag{60}$$

The key point of this step is that something like x_{n+1} which can be treated as a virtual input, does not appear in this error dynamics anymore. To negate the derivative of Lyapunov function, a suitable dynamics must be imposed to this error subsystem. Therefore, the following equation will be held for the n-th error dynamics

$$-a_0\hat{x}_1 + b_0 u - \dot{\alpha}_{n-1} = -c_n z_n - z_{n-1} + \alpha_{ns} \tag{61}$$

By this means the time derivative of the overall Lyapunov function

$$V = V_n = \frac{1}{2}\sum_{i=1}^{n} z_i^2 \tag{62}$$

can be computed as

$$\begin{aligned}\dot{V}_n = & -\sum_{i=1}^{n} c_i z_i^2 + z_1\left(\alpha_{1s} + \tilde{x}_2 - a_{n-1}\tilde{x}_1 + \Delta_1 + \dot{d}_y\right) \\ & + \sum_{i=2}^{n-1} z_i(\alpha_{is} + \tilde{x}_{i+1} - a_{n-i}\tilde{x}_1 + \Delta_i) \\ & + z_n(\alpha_{ns} - a_0\tilde{x}_1 + \Delta_n)\end{aligned} \tag{63}$$

Considering the conditions (38) and (39) for robust control α_{ns}, this *Lyapunov* function can satisfy the stability requirements for the overall system in spite of various uncertainties. The control input u can be obtained implicitly as the solution of the linear time-varying differential equation defined by (59) and (61). First we define

$$\varphi_i \triangleq a_{m-i}\hat{x}_1 + \dot{\alpha}_{(\rho+i-1)s} - c_{\rho+i}z_{\rho+i} - z_{\rho+i-1} \tag{64}$$

If we consider

$$\begin{aligned}&\xi_1 \triangleq \alpha_{\rho a} \\ &\xi_{i+1} \triangleq \alpha_{(\rho+i)a} \quad (1 \le i \le m-1)\end{aligned} \tag{65}$$

as state variables, we can write $(1 \le i \le m-1)$

$$\begin{aligned}\dot{\xi}_i &= \xi_{i+1} - \varphi_i + \frac{b_{m-i}}{b_m}\left(-\xi_1 + \varphi_0 + \dot{\alpha}_{(\rho-1)a}\right) \\ \dot{\xi}_m &= -\alpha_{ns} - \varphi_m + \frac{b_0}{b_m}\left(-\xi_1 + \varphi_0 + \dot{\alpha}_{(\rho-1)a}\right) \\ u &= \frac{1}{b_m}\left(-\xi_1 + \varphi_0 + \dot{\alpha}_{(\rho-1)a}\right)\end{aligned} \tag{66}$$

This u should be used as u_c as the overall control input.

V. Simulation Results

This section presents an example to illustrate how to implement the proposed procedure in practice, and to examine the effectiveness of the controller in comparison to that of a conventional robust controller. The models used in this example are based on a real experimental setup at the University of Toronto [12]. Consider the 4th order flexible beam manipulator with the following nominal model

$$P_0(s) = \frac{1.295(-s+5.531)(s+4.904)}{s^4 + 0.714s^3 + 27.90s^2 + 0.019s} \tag{67}$$

and uncertainty profile

$$W(s) = \frac{0.881(s^2 + 1.2s + 1)}{(s+0.001)(s+1.2)(s+1000)} \tag{68}$$

Notice that this system is non-minimum phase and has an unstable zero at 5.531 radians. Suppose that it is intended to design a suitable controller for this manipulator to track the reference trajectory y_r, shown in Fig. 3. The trajectory settles down in 5 seconds. Firs we choose the reference model

$$P_m(s) = \frac{1.295(s+5.531)(s+4.904)}{s^4 + 0.714s^3 + 27.90s^2 + 0.019s + 10^{-6}} \tag{69}$$

This model has the same poles, zeros and DC gain of the real system, and only the unstable zero of the original system is replace by a stable zero at the same frequency, and since, the reference model must be stable, a small term (10^{-6}) is added to the denumerator of P_m. Now apply the equations of adaptation part given in Section III. The adaptive routine will be completed by choosing the adaptation rate parameter as

$$\gamma = 10^{-4} \tag{70}$$

and the parameters initial values as

$$\theta^0 = [0 \quad 0 \quad 0 \quad 0 \quad 1]^T. \tag{71}$$

Next, the robust part should be designed. From Section IV the backstepping procedure is applied step by step. Since the relative degree of the system is two, the controller may be designed in only two steps to have a stable response. However, dynamic backstepping up to four step may be implemented to reach to more suitable performance. Here we stay on the 2nd step because the performance of backstepping controller is good enough. Finally, the controller parameters is fine tuned as following to achieve a good performance:

$$\begin{aligned} &c_1 = 2.8 \times 10^4, c_2 = 5.5 \times 10^2 \\ &r_1 = 70, r_2 = 20, h_1 = 2 \times 10^3, h_2 = 10^2 \end{aligned} \tag{72}$$

Now, simulate the designed controller on the perturbed system using the uncertainty profile given in (68). As it shown in Fig. 3 the output position of the manipulator well tracks the trajectory, and finally reaches to the final value without any steady state error. The tracking error is shown in the third figure row, which shows that the maximum tracking error occurs at transient response and is less than 0.1 radians. The input signals show that the robust portion of the controller, which produces the shown control signal u_c in figure, contributes to the main part of the output, and the adaptive part, which receives u_c as input and composes control signal u in order to force the behavior of system close to the reference model, is relatively smaller.

In order to show the significance of the proposed controller, the same system is considered using a conventional pure robust controller reported in [12].

In order to show the effectiveness of the proposed controller, the closed loop response is compared to that of the same perturbed system with a robust H∞ controller reported in [12]. The closed loop performances of both controllers are given in Fig. 3. As it is clearly seen the tracking performance of the proposed controller is farther better than that of the H∞ controller. The ARC response is much faster and more precise. Furthermore, its control effort of using this controller is much less and more implementable than that of the H∞ controller.

TABLE I compares the transient and steady state performance of these controllers using different measures, and clearly verifies that ARC controller has reached to smaller tracking errors in both transient and steady state, while the control effort is reduced significantly compared to that of the H∞ controller. Finally, the settling time of the proposed controller is in complete agreement with the required time trajectory. Therefore, the ARC controller outperforms conventional controller with a large margin.

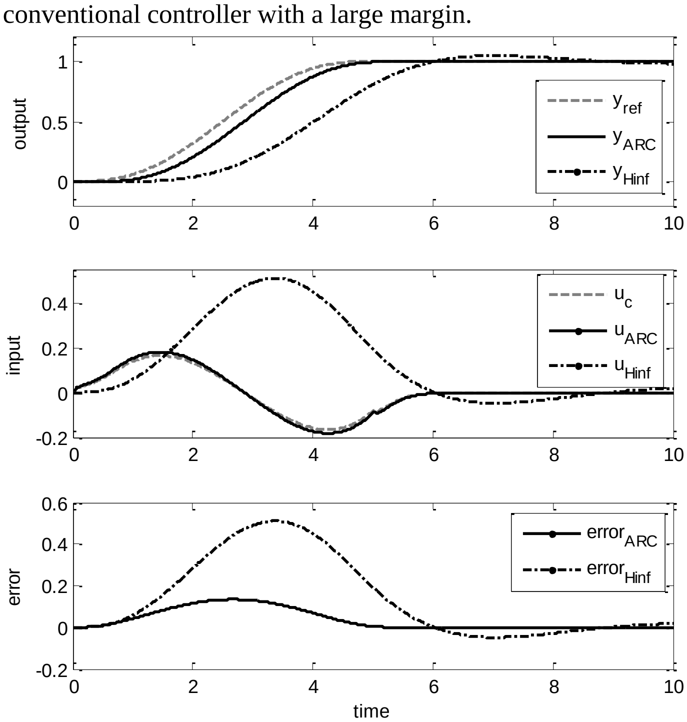

Fig. 3. The closed-loop response of a perturbed system with both the proposed ARC controller and a robust H∞ controller.

TABLE I
PERFORMANCE INDICES FOR COMPARING CONTROLLERS

Index	ARC Controller	H∞ Controller
Steady-state error (e_{ss})	2.12×10^{-5}	1.8×10^{-2}
Max. of abs. error ($\max\lvert e\rvert$)	0.134	0.510
Mean of error ($\lVert e\rVert$)	1.918	7.533
Settling time (t_s)	4.42	5.66
Control effort ($\lVert u\rVert$)	2.935	7.533

VI. CONCLUSIONS

In this paper, an alternative algorithm for synthesis of dynamical backstepping design procedure is proposed to develop an ARC controller of non-minimum phase systems. In this method, the dynamic backstepping controller ensures the robustness of the tracking error performance, while adaptation mechanism enforces to control the system to behave as the reference model. This procedure can significantly reduce the uncertainty, and consequently, this combination reduces the control effort exerted by the robust part of the controller. Moreover, the lack of stability for non-minimum phase systems is rectified using this adaptive procedure. Results of simulations executed for a flexible manipulator verifies the possible stable responses of non-minimum phase systems and shows the effectiveness of this structure in terms of transient and steady state performance, tracking errors, and disturbance rejection. Finally, comparing the results obtained by the proposed controller to that of a conventional robust H∞ controller shows that the proposed method significantly outperforms the latter.

REFERENCES

[1] B. Yao, M. Al-Majed and M. Tomizuka, “High-performance robust motion control of machine tools: An adaptive robust control approach and comparative experiments,” *IEEE/ASME Trans. Mechatronics*, vol. 2, no. 2, pp. 63-76, 1997.

[2] H. Taghirad and E. Jamei, “Robust Performance Verification of IDC ARC Controller for Hard Disk Drives,” *IEEE Trans. Industrial Electronics*, vol. 55, no. 1, pp. 448–456, Jan. 2008.

[3] J. Q. Gong and B. Yao, “Global stabilization of a class of uncertain systems with saturated adaptive robust controls,’’ in *IEEE Conf. on Decision and Control*, Sydney, 2000, pp. 1882-1887.

[4] L. Xu and B. Yao, “Output feedback adaptive robust control of uncertain linear systems with large disturbances.” in *Proc. of American Control Conference*, San Diego, 1999, pp. 556-560.

[5] A. S. Zinober, J. C. Scarratt, R. E. Mills, and A. J. Koshkouei, “New developments in dynamical adaptive backstepping control,” Berlin: Springer, 2001.

[6] Ye Jiang, Qinglei Hu, Guangfu Ma, “Adaptive backstepping fault-tolerant control for flexible spacecraft with unknown bounded disturbances and actuator failures,” *ISA Transactions*, vol. 49, no. 1, pp. 57-69, 2010.

[7] H. H. Eghrary, “Dynamic adaptive robust backstepping control for dual-stage hard disk drives,” M.S. thesis, Dept. Control Eng., K. N. Toosi Univ. of Tech., Tehran, Iran, 2009.

[8] M. Kristic, I. Kanellakopoulos and P. V. Kokotovic, “Nonlinear and adaptive control design,” New York: Willey, 1995.

[9] K.-J. Astrom, *Adaptive Control* (2nd Edition). Addison Wesley, 1995, ch 5.

[10] M. Kristic, I. Kanellakopoulos and P. V. Kokotovic, “Nonlinear design of adaptive controller for linear systems,” *IEEE Trans. on Automatic Control*, vol. 39, no. 4, pp. 738-752, 1994.

[11] B. Yao, “High performance adaptive robust control of nonlinear: A general framework and new scheme,” in *Proc. of 36th Conf. on Decision and Control*, December 1997, pp. 2489-94.

[12] J.-C. Doyle, B.-A. Francis and A.-R. Tannenbaum, *Feedback control theory*. Macmillan Publishing Co., 1990, ch 10.