Adaptive Regularization for a Class of Nonlinear Affine Differential-Algebraic Equation Systems

Zongtao Lu and Xiaoping Liu

Abstract— In this paper, the regularization problem is investigated for nonlinear differential-algebraic equation (DAE) systems with unknown parameters, which appear linearly in both differential and algebraic equations. It is shown that the feasibility of the proposed algorithms guarantees the existence of a feedback controller so that the resulting closed-loop systems admit equivalent ordinary differential equation (ODE) systems with lower triangular forms. As an application case of DAE systems, a constrained manipulator with flexible joints is studied to illustrate the proposed methodology.

I. INTRODUCTION

Differential-Algebraic Equation (DAE) systems (also referred to as singular, descriptor, semistate, generalized systems etc.) arise naturally as dynamic model of electrical [18], mechanical [17] and chemical engineering [3] applications. It embodies an important class of systems of both theoretical interest and practical significance. In theoretical research on DAE systems, the work focused on the issues related to solvability and numerical solutions [1] [2]. There was some work on the topics of passivity based design [19], feedback linearization [6], observer design [22], disturbance decoupling [13], input-output decoupling [14], regulation [5] [15], output tracking [7] [12], stabilization [16], robust stabilization [11], over- and under-determined nonlinear analysis [10]. In practical applications, it is known that mechanical systems with classical holonomic and nonholonomic constraints [21] and robotic systems with kinematic constraints [7] are modeled naturally by DAE systems. DAE systems are also known as dynamic models in power systems [4] and chemical processes [9].

In above mentioned theoretical research and practical applications, the parameters in the DAE systems are normally assumed to be known or given. However, it is not the case in general. For example, for a constrained robotic system, the parameters such as damping, stiffness and friction coefficients in the dynamic equations are normally unknown due to the difficulty of measurement. Therefore, to investigate the adaptive control problem of DAE systems becomes natural and significant.

In this paper, we consider a MIMO nonlinear affine DAE

system

$$\dot{x} = f_1(x) + p_1(x)z + g_1(x)u + \alpha_1(x)\theta \quad (1)$$
$$0 = f_2(x) + p_2(x)z + g_2(x)u + \alpha_2(x)\theta \quad (2)$$
$$y = h(x) \quad (3)$$

where $x \in R^n$, $z \in R^s$, $u \in R^m$, $y \in R^m$ and $\theta \in R^t$ are the vector of differential variables, algebraic variables, inputs, outputs, and unknown parameters respectively. $f_1(x)$, $f_2(x)$, $p_1(x)$, $p_2(x)$, $g_1(x)$, $g_2(x)$, $\alpha_1(x)$, $\alpha_2(x)$, $h(x)$ are matrix-valued smooth functions with dimensions of $n \times 1$, $s \times 1$, $n \times s$, $s \times s$, $n \times m$, $s \times m$, $n \times t$, $s \times t$, $m \times 1$, respectively. Assume that the origin is an isolated equilibrium point, i.e. $f_1(0) = 0$, $f_2(0) = 0$ and $h(0) = 0$. In this paper, we develop one methodology to regularize this class of DAE system, i.e., finding a change of coordinates to transform the DAE system into an equivalent ODE system with a lower triangular structure. A state feedback controller is constructed for the resulting system.

The proposed methodology consists of three algorithms. By repetitively calculating the so called generalized characteristic numbers given in Algorithm 1, Algorithm 2 is proposed to identify the constraints hidden behind the algebraic equations. These two algorithms can be considered as extensions of the first and second algorithm in [16]. In comparison with the algorithms in [16], one more term will be considered each step of our algorithms, which is due to $\alpha_2(x)$. Algorithm 3 constructs the change of coordinates. The key step of the change of coordinates involves the design of a static feedback $u = \gamma(x)z + v$. The resulting ODE system is in a lower triangular form and the design adaptive controller guarantees the global asymptotic stability of the closed-loop system if the change of coordinates is defined globally.

In the next section, we present the problem formulation and three developed algorithms. Based on the resulting ODE system with lower triangular form, an adaptive controller is designed by the backstepping technique in Section 3. As an important application of DAE systems, one constrained robotic system with flexible joints is studied in the Section 4. Concluding remarks are given in Section 5.

II. PROBLEM FORMULATION AND MAIN RESULTS

Considering the DAE system (1)-(3), the adaptive regularization problem of this system is to find a feedback $u = \gamma(x)z + \alpha(x, \widehat{\theta})$ and adaptive law $\dot{\widehat{\theta}} = \widehat{\theta}(x, \widehat{\theta})$, with $\gamma(x)$ and $\alpha(x, \widehat{\theta})$ smooth functions defined in a neighborhood U of the origin and $\alpha(0, \widehat{\theta}) = 0$, such that the corresponding

closed-loop system

$$\dot{x} = f_1(x) + g_1(x)\alpha(x, \widehat{\theta}) \tag{4}$$
$$+ [p_1(x) + g_1(x)\gamma(x)]z + \alpha_1(x)\theta$$
$$0 = f_2(x) + g_2(x)\alpha(x, \widehat{\theta}) \tag{5}$$
$$+ [p_2(x) + g_2(x)\gamma(x)]z + \alpha_2(x)\theta$$
$$y = h(x) \tag{6}$$

has the following properties:

1. for any consistent initial condition $x_0 \in U$, it has a unique differentiable solution $(x(t), z(t))$ with $x(0) = x_0$;

2. $\lim_{t \to \infty} x(t) = 0$ for any consistent initial condition $x_0 \in U$.

The following algorithm is given with the assumption as $\begin{bmatrix} p_2(x) & g_2(x) \end{bmatrix}$ has a full row rank.

Algorithm 1: *Calculation of the Generalized Characteristic Number*

Step 1. Assign $\phi_0(x) := \phi(x)$ and set $k = 0$. Calculate $L_{f_1}\phi_0(x)$, $L_{p_1}\phi_0(x)$, $L_{g_1}\phi_0(x)$ and $L_{\alpha_1}\phi_0(x)$. If the matrix $\begin{bmatrix} p_2(x) & g_2(x) \\ L_{p_1}\phi_0(x) & L_{g_1}\phi_0(x) \end{bmatrix}$ has a constant rank s, then there exists a unique vector-valued smooth function $e_0(x)$ of dimension s such that

$$\begin{bmatrix} L_{p_1}\phi_0(x) & L_{g_1}\phi_0(x) \end{bmatrix} = e_0(x) \begin{bmatrix} p_2(x) & g_2(x) \end{bmatrix}$$

Define that $\phi_1(x) = L_{f_1}\phi_0(x) - e_0(x)f_2(x)$ and $w_1(x) = L_{\alpha_1}\phi_0(x) - e_0(x)\alpha_2(x)$. Otherwise, set $r = 1$ and terminate the algorithm.

Step $k + 1$. Suppose we have already defined a sequence of $\phi_0(x)$, $\phi_1(x) \cdots \phi_k(x)$. Now calculate $L_{f_1}\phi_k(x)$, $L_{p_1}\phi_k(x)$, $L_{g_1}\phi_k(x)$ and $L_{\alpha_1}\phi_k(x)$. If the matrix $\begin{bmatrix} p_2(x) & g_2(x) \\ L_{p_1}\phi_k(x) & L_{g_1}\phi_k(x) \end{bmatrix}$ has a constant rank s, then there exists a unique vector-valued smooth function $e_k(x)$ of dimension s such that

$$\begin{bmatrix} L_{p_1}\phi_k(x) & L_{g_1}\phi_k(x) \end{bmatrix} = e_k(x) \begin{bmatrix} p_2(x) & g_2(x) \end{bmatrix}$$

Define that $\phi_{k+1}(x) = L_{f_1}\phi_k(x) - e_k(x)f_2(x)$ and $w_{k+1}(x) = L_{\alpha_1}\phi_k(x) - e_k(x)\alpha_2(x)$. Otherwise, set $r = k+1$ and terminate the algorithm.

The algorithm terminates at Step r. Such an integer is defined to be the generalized characteristic number of the function $\phi(x)$ under the constraint (2). Differentiating $\phi_k(x)$ with respect to time, it follows that for $k = 0, 1, \cdots, r - 2$

$$\frac{d\phi_k(x)}{dt} = L_{f_1}\phi_k(x) + L_{p_1}\phi_k(x)z + L_{g_1}\phi_k(x)u \tag{7}$$
$$+ L_{\alpha_1}\phi_k(x)\theta$$
$$= \phi_{k+1}(x) + w_{k+1}(x)\theta$$
$$+ e_k(x)[f_2(x) + p_2(x)z + g_2(x)u + \alpha_2(x)\theta]$$

and

$$\frac{d\phi_{r-1}(x)}{dt} = L_{f_1}\phi_{r-1}(x) + L_{p_1}\phi_{r-1}(x)z \tag{8}$$
$$+ L_{g_1}\phi_{r-1}(x)u + L_{\alpha_1}\phi_{r-1}(x)\theta$$

Remark 1: Algorithm 1 will be used in the following Algorithm 2 and 3. Different from the algorithms in [14] and

[16], besides ϕ_k, Algorithm 1 is also involved in calculating w_k due to the $\alpha_2(x)$ term, where θ enters.

Algorithm 2: *Regularization*

Algorithm 2 is used to identify all the hidden constraints behind the algebraic equation (2). It begins with decomposing the algebraic equation (2) into the form of (9) and (10). Each step of Algorithm 2 involves calculating the generalized characteristic number r^i of the function $\phi^i(x)$ under the constraint produced in the previous step. The algorithm is developed under the assumption that the hidden constraints on x is independent of θ.

Step 0. Consider the constraint (2) and suppose the matrix $\begin{bmatrix} p_2(x) & g_2(x) \end{bmatrix}$ has a constant rank s_0. Without loss of generality, assume that its first s_0 rows, denoted by $\begin{bmatrix} b_0(x) & c_0(x) \end{bmatrix}$, has a full row rank s_0. Let $p = s - s_0$, then for each $i = 1, 2, \cdots, p$, there exists a unique vector $S^i(x)$ such that

$$\begin{bmatrix} p_2^{s_0+i}(x) & g_2^{s_0+i}(x) \end{bmatrix} = S^i(x) \begin{bmatrix} b_0(x) & c_0(x) \end{bmatrix}$$

where $p_2^{s_0+i}(x)$ and $g_2^{s_0+i}(x)$ are the $(s_0+i)th$ row of $p_2(x)$ and $g_2(x)$ respectively.

Set $\phi^i(x) = f_2^{s_0+i}(x) - S^i(x)a_0(x)$ and $w^i(x) = \alpha_2^{s_0+i}(x) - S^i(x)d_0(x)$ with $a_0(x)$, $d_0(x)$ being the first s_0 rows of $f_2(x)$, $\alpha_2(x)$ respectively and $f_2^{s_0+i}(x)$, $\alpha_2^{s_0+i}(x)$ being the first $(s_0 + i)th$ rows of $f_2(x)$, $\alpha_2(x)$ respectively. Then the algebraic equation (2) becomes

$$0 = a_0(x) + b_0(x)z + c_0(x)u + d_0(x)\theta \tag{9}$$
$$0 = \phi^i(x) + w^i(x)\theta + S^i(x)[a_0(x) \tag{10}$$
$$+ b_0(x)z + c_0(x)u + d_0(x)\theta]$$

for $i = 1, 2, \cdots, p$.

Substituting (9) into (10) leads to

$$0 = \phi^i(x) + w^i(x)\theta \tag{11}$$

Considering that the hidden constraints on x is independent of θ, we have $w^i(x) \equiv 0$ for $i = 1, 2, \cdots, p$. So the hidden constraint (11) becomes

$$0 = \phi^i(x) \tag{12}$$

Step 1. Assign $\phi(x) := \phi^1(x)$ and carry out Algorithm 1 to calculate its generalized characteristic number under the constraint (9). Then r^1, $\phi_0^1(x)$, $\phi_1^1(x)$, $\cdots$, $\phi_{r^1-1}^1(x)$, $w_1^1(x)$, $\cdots$, $w_{r^1-1}^1(x)$ and $e_0^1(x)$, $e_1^1(x)$, $\cdots$, $e_{r^1-2}^1(x)$ are produced. Now define $a_1(x) = L_{f_1}\phi_{r^1-1}^1(x)$, $b_1(x) = L_{p_1}\phi_{r^1-1}^1(x)$, $c_1(x) = L_{g_1}\phi_{r^1-1}^1(x)$ and $d_1(x) = L_{\alpha_1}\phi_{r^1-1}^1(x)$.

Differentiating $\phi_i^1(x)$ with respect to time leads to for $i = 0, 1, \cdots, r^1 - 2$

$$\frac{d\phi_i^1(x)}{dt} = \phi_{i+1}^1(x) + w_{i+1}^1(x)\theta + e_i^1(x)[a_0(x) \tag{13}$$
$$+ b_0(x)z + c_0(x)u + d_0(x)\theta]$$

and

$$\frac{d\phi_{r^1-1}^1(x)}{dt} = a_1(x) + b_1(x)z + c_1(x)u + d_1(x)\theta \tag{14}$$

It follows from (12) that $\frac{d\phi_0^1(x)}{dt} = 0$, from which, together with (13), we obtain the hidden constraint $0 = \phi_1^1(x) + w_1^1(x)\theta$.

Considering that the hidden constraints on x is independent of θ, we have $w_1^1(x) = 0$ and the new hidden constraint is $0 = \phi_1^1(x)$. By the same token, it can be derived that $w_i^1(x) \equiv 0$ and

$$\phi_i^1(x) = 0 \qquad (15)$$

for $i = 1, 2, \cdots, r^1 - 1$. It follows from (15) with $i = r^1 - 1$ and (14) that

$$0 = a_1(x) + b_1(x)z + c_1(x)u + d_1(x)\theta \qquad (16)$$

Combining (9) and (16) yields the following algebraic equation

$$0 = a^1(x) + b^1(x)z + c^1(x)u + d^1(x)\theta \qquad (17)$$

where $a^1(x) = \begin{bmatrix} a_0(x) \\ a_1(x) \end{bmatrix}$, $b^1(x) = \begin{bmatrix} b_0(x) \\ b_1(x) \end{bmatrix}$, $c^1(x) = \begin{bmatrix} c_0(x) \\ c_1(x) \end{bmatrix}$, $d^1(x) = \begin{bmatrix} d_0(x) \\ d_1(x) \end{bmatrix}$.

If the matrix $\begin{bmatrix} b^1(x) & c^1(c) \end{bmatrix}$ has a full row rank $s_0 + 1$, then set $k = 2$ and go to next step. Otherwise, terminate the algorithm.

Step k. Suppose the algebraic equation produced at Step $k - 1$ is given by

$$0 = a^{k-1}(x) + b^{k-1}(x)z + c^{k-1}(x)u + d^{k-1}(x)\theta \qquad (18)$$

Assign $\phi(x) := \phi^k(x)$ and carry out Algorithm 1 to calculate its generalized characteristic number under the constraint (18). Then r^k, $\phi_0^k(x)$, $\phi_1^k(x)$, $\cdots$, $\phi_{r^k-1}^k(x)$, $w_1^k(x)$, $\cdots$, $w_{r^k-1}^k(x)$ and $e_0^k(x)$, $e_1^k(x)$, $\cdots$, $e_{r^k-2}^k(x)$ are produced. Now define $a_k(x) = L_{f_1}\phi_{r^k-1}^k(x)$, $b_k(x) = L_{p_1}\phi_{r^k-1}^k(x)$, $c_k(x) = L_{g_1}\phi_{r^k-1}^k(x)$ and $d_k(x) = L_{\alpha_1}\phi_{r^k-1}^k(x)$.

Differentiating $\phi_i^k(x)$ with respect to time produces

$$\frac{d\phi_i^k(x)}{dt} = \phi_{i+1}^k(x) + w_{i+1}^k(x)\theta + e_i^k(x)[a^{k-1}(x) \quad (19)$$
$$+ b^{k-1}(x)z + c^{k-1}(x)u + d^{k-1}(x)\theta]$$

for $i = 0, 1, \cdots, \ldots, r^k - 2$ and

$$\frac{d\phi_{r^k-1}^k(x)}{dt} = a_k(x) + b_k(x)z + c_k(x)u + d_k(x)\theta \qquad (20)$$

It follows from (12) that $\frac{d\phi_0^k(x)}{dt} = 0$, from which, together with (19), we obtain the hidden constraint $0 = \phi_1^k(x) + w_1^k(x)\theta$. Considering that the hidden constraints on x is independent of θ, we have $w_1^k(x) = 0$ and the new hidden constraint is $0 = \phi_1^k(x)$. By the same token, it can be derived that $w_i^k(x) \equiv 0$ and

$$\phi_i^k(x) = 0 \qquad (21)$$

for $i = 1, 2, \cdots, r^k - 1$. It follows from (21) with $i = r^k - 1$ and (20) that

$$0 = a_k(x) + b_k(x)z + c_k(x)u + d_k(x)\theta \qquad (22)$$

Combining (18) and (22) yields the following algebraic equation

$$0 = a^k(x) + b^k(x)z + c^k(x)u + d^k(x)\theta \qquad (23)$$

where $a^k(x) = \begin{bmatrix} a^{k-1}(x) \\ a_k(x) \end{bmatrix}$, $b^k(x) = \begin{bmatrix} b^{k-1}(x) \\ b_k(x) \end{bmatrix}$, $c^k(x) = \begin{bmatrix} c^{k-1}(x) \\ c_k(x) \end{bmatrix}$, $d^k(x) = \begin{bmatrix} d^{k-1}(x) \\ d_k(x) \end{bmatrix}$.

If the matrix $\begin{bmatrix} b^k(x) & c^k(c) \end{bmatrix}$ has a full row rank $s_0 + k$, then set $k = k + 1$ and go to next step. Otherwise, terminate the algorithm.

Algorithm 2 is said to be *feasible* if it terminates at Step $k = p$ and the matrix $\begin{bmatrix} b^p(x) & c^p(c) \end{bmatrix}$ has a full row rank $s = s_0 + p$. It follows from Algorithm 2 that in order for solutions to the DAE system (1)-(2) to be impulse-free, the initial condition $x(0)$ must satisfy $x(0) \in M$ with

$$M = \{x \in R^n \mid \phi_0^i(x) = 0, \phi_j^i(x) = 0, \text{ and } w_j^i = 0,$$
$$\text{for } j = 1, 2, \cdots, r^{i-1}, i = 1, 2, \cdots, p\}$$

If Algorithm 2 is feasible, the DAE system (1)-(2) is equivalent to the following DAE system

$$\dot{x} = f_1(x) + p_1(x)z + g_1(x)u + \alpha_1(x)\theta \qquad (24)$$

$$0 = a(x) + b(x)z + c(x)u + d(x)\theta \qquad (25)$$

where $x \in M$ and $\begin{bmatrix} b(x) & c(x) \end{bmatrix}$ has a full row rank s.

With the assumption that Algorithms 1 and 2 are feasible, the DAE system (24)-(25) can be changed to lower triangular form by a feedback controller $u = \gamma(x)z + v$ and the following algorithm.

Algorithm 3: *Standardization*

Algorithm 3 is applied to transform DAE systems into equivalent ODE systems. Algorithm 1 is used repetitively in Algorithm 3.

Step 1. Set $\psi^1(x) := h^1(x)$ and calculate the generalized characteristic number q^1 of $\psi^1(x)$ under the constraint (25). Then q^1, $\psi_0^1(x)$, $\psi_1^1(x)$, $\cdots$, $\psi_{q^1-1}^1(x)$, $\varphi_1^1(x)$, $\cdots$, $\varphi_{q^1-1}^1(x)$ and $E_0^1(x)$, $E_1^1(x)$, $\cdots$, $E_{q^1-2}^1(x)$ are produced. Now define $A_1(x) = L_{f_1}\psi_{q^1-1}^1(x)$, $B_1(x) = L_{p_1}\psi_{q^1-1}^1(x)$, $C_1(x) = L_{g_1}\psi_{q^1-1}^1(x)$ and $D_1(x) = L_{\alpha_1}\phi_{q^1-1}^1(x)$.

Differentiating $\psi_i^1(x)$ with respect to time yields

$$\frac{d\psi_i^1(x)}{dt} = \psi_{i+1}^1(x) + \varphi_{i+1}^1(x)\theta + E_i^1(x)[a(x) \quad (26)$$
$$+ b(x)z + c(x)u + d(x)\theta]$$

for $i = 0, 1, \cdots, \ldots, q^1 - 2$ and

$$\frac{d\psi_{q^1-1}^1(x)}{dt} = A_1(x) + B_1(x)z + C_1(x)u + D_1(x)\theta \quad (27)$$

Step k. Set $\psi^k(x) := h^k(x)$ and calculate the generalized characteristic number q^k of $\psi^k(x)$ under the constraint (25). Then q^k, $\psi_0^k(x)$, $\psi_1^k(x)$, $\cdots$, $\psi_{q^k-1}^k(x)$, $\varphi_1^k(x)$, $\cdots$, $\varphi_{q^k-1}^k(x)$ and $E_0^k(x)$, $E_1^k(x)$, $\cdots$, $E_{q^k-2}^k(x)$ are produced. Now define $A_k(x) = L_{f_1}\psi_{q^k-1}^1(x)$, $B_k(x) = L_{p_1}\psi_{q^k-1}^1(x)$, $C_1(x) = L_{g_1}\psi_{q^k-1}^1(x)$ and $D_1(x) = L_{\alpha_1}\phi_{q^k-1}^1(x)$.

Differentiating $\psi_i^k(x)$ with respect to time leads to for $i = 0, 1, \cdots, ..., q^k - 2$

$$\frac{d\psi_i^k(x)}{dt} = \psi_{i+1}^k(x) + \varphi_{i+1}^k(x)\theta + E_i^k(x)[a(x) \quad (28)$$
$$+ b(x)z + c(x)u + d(x)\theta]$$

and

$$\frac{d\psi_{q^k-1}^k(x)}{dt} = A_k(x) + B_k(x)z + C_k(x)u + D_k(x)\theta \quad (29)$$

Algorithm 3 terminates at Step $k = m$. Now the following assumptions are made.

Assumption 1: The matrix $\begin{bmatrix} b(x) & c(x) \\ B_1(x) & C_1(x) \\ \vdots & \vdots \\ B_m(x) & C_m(x) \end{bmatrix}$ is non-singular in U.

Assumption 2: $n = r + q$ with $r = r^0 + r^1 + \cdots + r^p$ and $q = q^0 + q^1 + \cdots + q^m$.

The functions $\phi_j^i(x)$ for $j = 0, 1, \cdots, ..., r^i - 1$ and $i = 0, 1, \cdots, ..., p$, and $\psi_j^i(x)$ for $j = 0, 1, \cdots, ..., q^i - 1$ and $i = 0, 1, \cdots, ..., m$, form a set of new coordinates, which is guaranteed by Lemma 1, for proof, see [16].

Lemma 1: Suppose that the algorithms 1-3 are feasible and Assumption 1 is satisfied. Then, the vectors

$$d\phi_0^1(x), d\phi_1^1(x), \cdots, d\phi_{r^1-1}^1(x)$$
$$\vdots$$
$$d\phi_0^p(x), d\phi_1^p(x), \cdots, d\phi_{r^p-1}^p(x)$$
$$d\psi_0^1(x), d\psi_1^1(x), \cdots, d\psi_{q^1-1}^1(x)$$
$$\vdots$$
$$d\psi_0^m(x), d\psi_1^m(x), \cdots, d\psi_{q^m-1}^m(x)$$

are linearly independent in U.

Remark 2: Assumption 2 is introduced to avoid the appearance of zero dynamics. It is needed to point out that, as a matter of fact, similar conclusions can also be obtained for the case $r + q < n$ if the appearing zero dynamics is asymptotically stable in U.

With Assumption 2, it follows from Lemma 1 that the function $\Phi(x) = \begin{bmatrix} \phi(x) \\ \psi(x) \end{bmatrix}$ constitutes a change of coordinates, where $\phi(x) = \begin{bmatrix} \phi^1(x)^T & \phi^2(x)^T & \cdots & \phi^p(x)^T \end{bmatrix}^T$ and $\psi(x) = \begin{bmatrix} \psi^1(x)^T & \psi^2(x)^T & \cdots & \psi^m(x)^T \end{bmatrix}^T$ with $\phi^i(x) = \begin{bmatrix} \phi_0^i(x) & \cdots & \phi_{r^i-1}^i(x) \end{bmatrix}^T$ for $i = 1, \cdots, p$ and $\psi^j(x) = \begin{bmatrix} \psi_0^j(x) & \cdots & \psi_{q^j-1}^j(x) \end{bmatrix}^T$ for $j = 1, \cdots, m$.

Set $\varepsilon_i^k = \phi_i^k(x)$ for $i = 0, 1, \cdots, r^k - 1$, $k = 1, \cdots, p$ and $\xi_i^k = \psi_i^k(x)$ for $i = 0, 1, \cdots, q^k - 1$, $k = 1, \cdots, m$. Let $\varepsilon = \begin{bmatrix} \varepsilon_0^1 & \cdots & \varepsilon_{r^1-1}^1 & \cdots & \varepsilon_0^p & \cdots & \varepsilon_{r^p-1}^p \end{bmatrix}^T$ and $\xi = \begin{bmatrix} \xi_0^1 & \cdots & \xi_{q^1-1}^1 & \cdots & \xi_0^m & \cdots & \xi_{q^m-1}^m \end{bmatrix}^T$.

By differentiating ε_i^k and ξ_i^k with respect to time, in the new coordinates the DAE system (1)-(3) can be expressed as follows

$$\varepsilon = 0 \quad (30)$$

$$y^i = \xi_0^i$$
$$\dot{\xi}_0^i = \xi_1^i + \varphi_1^i(x)\theta$$
$$\vdots \quad (31)$$
$$\dot{\xi}_{q^i-2}^i = \xi_{q^i-1}^i + \varphi_{q^i-1}^i(x)\theta$$
$$\dot{\xi}_{q^i-1}^i = A_i(x) + B_i(x)z + C_i(x)u + D_i(x)\theta$$

where $i = 1, \cdots, m$.

In order to apply the adaptive backstepping technique, the following assumption is needed to transform the system (30) into lower triangular form.

Assumption 3: The matrix W in (45) has a constant row rank $r + \sum_{j=1}^{k-1} q^j + i + 1$ for $i = 0, 1, \cdots, q^k - 1$ and $k = 1, \cdots, m$.

Lemma 2: Suppose that Algorithm 1-3 are feasible and Assumption 1-3 are satisfied. Then, in ξ coordinates, the system (30)-(31) takes the form of

$$\varepsilon = 0 \quad (32)$$

$$y^i = \xi_0^i$$
$$\dot{\xi}_0^i = \xi_1^i + \varphi_1^i(\varepsilon, \xi_0^i)\theta$$
$$\vdots \quad (33)$$
$$\dot{\xi}_{q^i-2}^i = \xi_{q^i-1}^i + \varphi_{q^i-1}^i(\varepsilon, \xi_0^i, \xi_1^i, \cdots, \xi_{q^i-2}^i)\theta$$
$$\dot{\xi}_{q^i-1}^i = A_i(x) + B_i(x)z + C_i(x)u + D_i(x)\theta$$

where $i = 1, \cdots, m$.

The proof is given in Appendix.

Theorem 1: Consider the system (1)-(3). Suppose that Algorithm 1-3 are feasible and Assumption 1-3 are satisfied. There exist a feedback controller $u = \gamma(x)z + \alpha(x, \widehat{\theta})$ and an estimator $\dot{\widehat{\theta}} = \widehat{\theta}(x, \widehat{\theta})$ such that the corresponding closed-loop system is asymptotically stable in a neighborhood U of the origin.

A constructive proof is given in Section III.

III. DESIGN OF ADAPTIVE FEEDBACK CONTROLLER

We know that the matrix $\begin{bmatrix} b(x) & c(x) \end{bmatrix}$ is of full row rank from Algorithm 2, hence there exists a smooth matrix-valued function $\gamma(x)$ such that $b(x) + c(x)\gamma(x)$ is non-singular. By introducing a feedback $u = \gamma(x)z + v$, the algebraic equation (25) admits $a(x) + [b(x) + c(x)\gamma(x)]z + c(x)v + d(x)\theta = 0$ and solving it gives $z = -[b(x) + c(x)\gamma(x)]^{-1}[a(x) + c(x)v + d(x)\theta]$. As a result, u can be expressed as

$$u = -\gamma(x)[b(x) + c(x)\gamma(x)]^{-1}[a(x) + c(x)v + d(x)\theta] + v \quad (34)$$

Substituting u and z leads to

$$\varepsilon = 0 \quad (35)$$

$$y^i = \xi_0^i$$
$$\dot{\xi}_0^i = \xi_1^i + \varphi_1^i(0, \xi_0^i)\theta$$
$$\vdots \quad (36)$$
$$\dot{\xi}_{q^i-2}^i = \xi_{q^i-1}^i + \varphi_{q^i-1}^i(0, \xi_0^i, \xi_1^i, \cdots, \xi_{q^i-2}^i)\theta$$
$$\dot{\xi}_{q^i-1}^i = \widehat{V}_i + \varphi_{q^i}^i(0, \xi^1, \cdots, \xi^{m-1}, \xi^m)\theta$$

where $\hat{V}_i = A_i - [B_i + C_i\gamma(x)][b(x) + c(x)\gamma(x)]^{-1}a(x) + \{C_i - [B_k + C_i\gamma(x)][b(x)+c(x)\gamma(x)]^{-1}c(x)\}v$ and $\varphi_{q^i}^i(x) = [B_i + C_i\gamma(x)][b(x) + c(x)\gamma(x)]^{-1}d(x) + D_i$, for $i = 1, \cdots, m$.

By applying adaptive backstepping technique to design an adaptive controller, the system (35)-(36) is guaranteed to be globally asymptotically stable for any unknown parameters. The design procedure is routine and can be found in [8].

Suppose the designed feedback controller

$$\hat{V}_i = \hat{V}_{q^i}^i(\xi^1, \xi^2, \cdots, \xi^m, \hat{\theta}) \qquad (37)$$

and the parameter estimator

$$\dot{\hat{\theta}} = \hat{\theta}(\xi^1, \xi^2, \cdots, \xi^m) \qquad (38)$$

guarantee the asymptotic stability of the closed-loop system in a neighborhood U of the origin. With the controller (37) and the parameter estimator (38) in ξ coordinates, we will express them in the original x coordinates. For convenience, let $\hat{A}(x) = \begin{bmatrix} \hat{A}_1^T & \cdots & \hat{A}_m^T \end{bmatrix}^T$ with $\hat{A}_k(x) = A_k - [B_k + C_k\gamma(x)][b(x) + c(x)\gamma(x)]^{-1}a(x)$, $\hat{C}(x) = \begin{bmatrix} \hat{C}_1^T & \cdots & \hat{C}_m^T \end{bmatrix}^T$ with $\hat{C}_k = C_k - [B_k + C_k\gamma(x)][b(x) + c(x)\gamma(x)]^{-1}c(x)$, $\hat{V} = \begin{bmatrix} \hat{V}_1^T & \cdots & \hat{V}_m^T \end{bmatrix}^T$ for $k = 1, \cdots, m$. Hence, the controller u in the original coordinates is uniquely determined as

$$u = \gamma(x)z + \hat{C}^{-1}(x)[\hat{V} - \hat{A}(x)] \qquad (39)$$

With the following equation and Assumption 2, $\hat{C}(x)$ in (39) is guaranteed to be invertible since $b(x) + c(x)\gamma(x)$ is nonsingular.

$$\begin{bmatrix} b(x) + c(x)\gamma(x) & 0 \\ B_1(x) + C_1(x)\gamma(x) & \hat{C}_1 \\ \vdots & \vdots \\ B_m(x) + C_m(x)\gamma(x) & \hat{C}_m \end{bmatrix} = \begin{bmatrix} b(x) & c(x) \\ B_1(x) & C_1(x) \\ \vdots & \vdots \\ B_m(x) & C_m(x) \end{bmatrix} \begin{bmatrix} I & 0 \\ \gamma(x) & I \end{bmatrix} \begin{bmatrix} I & R \\ 0 & I \end{bmatrix}$$

with $R = -[b(x) + c(x)\gamma(x)]^{-1}c(x)$.

IV. EXAMPLE

A constrained robotic system will be studied as an example to illustrate the developed method in this section. Consider a constrained two-link robotic manipulator with two flexible joints [20] and suppose that the end-effector is in contact with a straight line constraint. Its dynamic model is expressed as

$$M(q)\ddot{q} + \beta(q, \dot{q}) + g(q) + Kq - K\theta = b(q)\lambda \qquad (40)$$
$$R\ddot{\theta} + D\dot{\theta} - Kq + K\theta = u \qquad (41)$$
$$\phi(q) = 0 \qquad (42)$$

where $q = (q_1, q_2)^T$ contains the link angles, $\theta = (\theta_1, \theta_2)^T$ the rotor angles, and $u = (u_1, u_2)^T$ the two inputs to the joint motors. R, K and D are the inertia matrix of joint motors,

the matrix of spring stiffness and the joint friction coefficient matrix respectively, with the value $R = diag(1, 1)$, $K = diag(100, 100)$ and $D = diag(d, d)$.

The inertia matrix is $M(q) = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}$ where $M_{11} = (l_2)^2 m_2 + 2l_1 l_2 m_2 \cos(q_2) + (l_1)^2(m_1 + m_2)$, $M_{12} = M_{21} = (l_2)^2 m_2 + l_1 l_2 m_2 \cos(q_2)$, $M_{22} = (l_2)^2 m_2$. The Coriolis, centrifugal and gravity terms are combined as $\beta(q, \dot{q}) + g(q) = \begin{bmatrix} \beta_1 & \beta_2 \end{bmatrix}^T$ with $\beta_1 = -m_2 l_1 l_2 \dot{q}_2(2\dot{q}_1 + \dot{q}_2)\sin(q_2) + gm_2 l_2 \cos(q_1 + q_2) + gl_1(m_1 + m_2)\cos(q_1)$, $\beta_2 = m_2 l_1 l_2 (\dot{q}_1)^2 \sin(q_2) + gm_2 l_2 \cos(q_1 + q_2)$, where the parameter values are $l_1 = l_2 = 0.3$ m, $m_1 = m_2 = 1$ kg, and $g = 9.8$ m/s^2.

Its Jacobian matrix is given by $b(q) = \begin{bmatrix} b_1 & b_2 \end{bmatrix}^T$ with $b_1 = l_1(\cos(q_1) + A\sin(q_1)) + l_2(\cos(q_1 + q_2) + A\sin(q_1 + q_2))$ and $b_2 = l_2(\cos(q_1 + q_2) + A\sin(q_1 + q_2))$.

The constraint is assumed to be a straight line described as $\phi(q) = y - Ax - B$ with $A = -1$, $B = 0.28$ and can be rewritten as $\phi(q) = l_1(\sin(q_1) - A\cos(q_1)) + l_2(\sin(q_1 + q_2) - A\cos(q_1 + q_2)) - B = 0$.

Let $x_1 = q_1, x_2 = q_2, x_3 = \dot{q}_1, x_4 = \dot{q}_1, x_5 = \theta_1, x_6 = \theta_2, x_7 = \dot{\theta}_1, x_8 = \dot{\theta}_2$ and $z = \lambda$. The system (40)- (42) can be written as

$$\dot{x}_1 = x_3$$
$$\dot{x}_2 = x_4$$
$$\begin{bmatrix} \dot{x}_3 & \dot{x}_4 \end{bmatrix}^T = M^{-1}(\cdot)N(\cdot) + M^{-1}(\cdot)b(\cdot)z$$
$$\dot{x}_5 = x_7$$
$$\dot{x}_6 = x_8$$
$$\dot{x}_7 = u_1 + 100x_1 - 100x_5 - dx_7$$
$$\dot{x}_8 = u_2 + 100x_2 - 100x_6 - dx_8$$
$$0 = \phi(x_1, x_2)$$

where $N(\cdot) = -\beta(x_1, x_2, x_3, x_4) - g(x_1, x_2) - 100\begin{bmatrix} x_1 & x_2 \end{bmatrix}^T + 100\begin{bmatrix} x_5 & x_6 \end{bmatrix}^T$.

Performing Algorithm 2 on $0 = \phi(x_1, x_2)$ gives $r^1 = 2$, $\phi_0(x_1, x_2) = l_1(\sin(q_1) - A\cos(q_1)) + l_2(\sin(q_1 + q_2) - A\cos(q_1 + q_2)) - B$, $\phi_1(x_1, x_2) = l_1(x_3\cos(x_1) + Ax_3\sin(x_1)) + l_2((x_3 + x_4)\cos(x_1 + x_2) + A(x_3 + x_4)\sin(x_1 + x_2))$, $a(x) = \Phi_1 + \Phi_2 M^{-1}(\cdot)N(\cdot)$, $b(x) = \Phi_2 M^{-1}(\cdot)b(x_1, x_2)$, $c = 0$, and $d = 0$ where $\Phi_1 = -l_1(x_3)^2\sin(x_1) + Al_1(x_3)^2\cos(x_1) - l_2(x_3 + x_4)^2\sin(x_1 + x_2) + Al_2(x_3 + x_4)^2\cos(x_1 + x_2)$ and $\Phi_2 = [\Phi_{21}, \Phi_{22}]$ with $\Phi_{21} = l_1\cos(x_1) + Al_2\sin(x_1) + l_2\cos(x_1 + x_2) + Al_2\sin(x_1 + x_2)$ and $\Phi_{22} = l_2\cos(x_1 + x_2) + Al_2\sin(x_1 + x_2)$.

Solving algebraic equations $0 = \phi_0(x_1, x_2)$ and $0 = \phi_1(x_1, x_2)$ for x_1 and x_3 gives $x_1 = P(x_2)$ and $x_3 = Q(x_1, x_2, x_4)$ where

$$P(\cdot) = \arcsin\left(\frac{a'B - b'\sqrt{(a')^2 + (b')^2 - B^2}}{((a')^2 + (b')^2)}\right)$$
$$Q(\cdot) = -x_4(l_2\cos(x_1 + x_2) + Al_2\sin(x_1 + x_2)) / (l_1\cos(x_1) + Al_1\sin(x_1) + l_2\cos(x_1 + x_2) + Al_2\sin(x_1 + x_2))$$

$$W = \begin{bmatrix} \frac{\partial\phi(x)}{\partial x} & \frac{\partial\psi_0^1(x)}{\partial x} & \cdots & \frac{\partial\psi_{q^1-1}^1(x)}{\partial x} & \cdots & \frac{\partial\psi_0^k(x)}{\partial x} & \cdots & \frac{\partial\psi_i^k(x)}{\partial x} & \frac{\partial\varphi_i^k(x)}{\partial x} \end{bmatrix}^T \quad (45)$$

$$U = \begin{bmatrix} \frac{\partial\phi(x)}{\partial\varepsilon} & \frac{\partial\psi_0^1(x)}{\partial\varepsilon} & \cdots & \frac{\partial\psi_{q^1-1}^1(x)}{\partial\varepsilon} & \cdots & \frac{\partial\psi_0^k(x)}{\partial\varepsilon} & \cdots & \frac{\partial\psi_i^k(x)}{\partial\varepsilon} & \frac{\partial\varphi_i^k(x)}{\partial\varepsilon} \\ \frac{\partial\phi(x)}{\partial\xi} & \frac{\partial\psi_0^1(x)}{\partial\xi} & \cdots & \frac{\partial\psi_{q^1-1}^1(x)}{\partial\xi} & \cdots & \frac{\partial\psi_0^k(x)}{\partial\xi} & \cdots & \frac{\partial\psi_i^k(x)}{\partial\xi} & \frac{\partial\varphi_i^k(x)}{\partial\xi} \end{bmatrix}^T \quad (46)$$

with $a' = l_1 + l_2\cos(x_2) + Al_2\sin(x_2)$ and $b' = -Al_1 + l_2\sin(x_2) - Al_2\cos(x_2)$. And solving the algebraic equation $0 = a(x) + b(x)z$ yields $z = -b^{-1}(x)a(x)$. By substituting z, x_1 and x_3 into the original system, the standard ODE system (43) is obtained, which is in the lower triangular form.

$$\begin{aligned}
\dot{x}_2 &= x_4 \\
\dot{x}_4 &= \chi_{21}(x_1,x_2)x_5 + \chi_{22}(x_1,x_2)x_6 + \kappa_2(x_1,x_2,x_3,x_4) \\
\dot{x}_5 &= x_7 \\
\dot{x}_6 &= x_8 \\
\dot{x}_7 &= u_1 + 100x_1 - 100x_5 - dx_7 \\
\dot{x}_8 &= u_2 + 100x_2 - 100x_6 - dx_8 \quad (43)
\end{aligned}$$

where

$$\begin{bmatrix} \chi_{11} & \chi_{12} \\ \chi_{21} & \chi_{22} \end{bmatrix} = 100M^{-1}(\cdot)[I - b(x_1,x_2)P]$$

$$\begin{bmatrix} \kappa_1 & \kappa_2 \end{bmatrix}^T = M^{-1}(\cdot)[-\beta(\cdot) - Q]$$

with $P = [\Phi_2(\cdot)M^{-1}(\cdot)b(x_1,x_2)]^{-1}\Phi_2(\cdot)M^{-1}(\cdot)$ and $Q = g(x_1,x_2) - 100\begin{bmatrix} x_1 & x_2 \end{bmatrix}^T$.

V. Conclusion

The adaptive regularization problem has been solved for a class of nonlinear affine DAE systems. Our methodology transforms the original system into one equivalent ODE system with lower triangular form. The designed adaptive feedback controller guarantees the global asymptotic stability if the change of coordinates is defined globally. One example of a constrained robotic system is illustrated for the proposed methodology.

References

[1] N.M. Ascher and L.R. Petzold, *Computer Methods for Ordinary Equations and Differential-Algebraic Equations*, New York, 1998.
[2] K.E. Brenan, S.L. Campbell, and L.R. Petzold, *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*, New York, 1996.
[3] G.D. Byrne and P.R. Ponzi, "Differential-Algebraic Systems, Their Applications and Solutions", *Comput. Chem. Engng.*, vol. 4, 1988, pp 377-382.
[4] D.J. Hill and I.M.Y. Mareels, "Stability Theory for Differential-Algebraic Systems with Applications to Power Systems", *IEEE Trans. Circuits and Systems*, vol. 37, 1990,1461-1423.
[5] J. Huang and J.F. Zhang, "Impulse-Free Output Regulation of Singular nonlinear Systems", *Int. J. of Control*, vol. 71, 1998, pp 789-806.
[6] S. Kawaji and E.Z. Taha, "Feedback Linearization of a Class of Nonlinear Descriptor Systems, *Proceedings of the IEEE Conference on Decision and Control*, pp 4035-4042, 1994.
[7] H Krishnan and N.H. McClamroch, "Tracking in Nonlinear Differential-Algebraic Control Systems with Applications to Contrained Robot Systems", *Automatica*, vol. 30, 1994, pp 885-897.
[8] M. Krstic, P.V. Kokotovic and I. Kanellakopoulos, *Nonlinear and Adaptive Control Design*, 1995, John Wiley & Sons, New York.
[9] A. Kumar and P. Daoutidis, *Control of Nonlinear Differential Algebraic Equations System with Applications to Chemical Process*, CHAPMAN & HALL/CRC, New York, 1999.
[10] P. Kunkel and V. Mehrmann, "Analysis of Over- and Underdetermined Nonlinear Differential-Algebraic Systems with Application to Nonlinear Control Problems", *Mathematics of Control, Signals, and Systems (MCSS)*, vol. 14, 2001, pp 233-256.
[11] X.P. Liu. "Robust Stabilization of Nonlinear Singular Systems", *Proc. of IEEE Conference on Decision and Control*, pp 2375-2376, 1995.
[12] X.P. Liu. "Asymptotic Output Tracking of Nonlinear Differential-Algebraic Control Systems", *Automatica*, vol. 34, 1998, pp 393-397.
[13] X.P. Liu. "Local Disturbance Decoupling of Nonlinear Differential-Algebraic Control Systems", *Int. J of Control*, vol. 70, 1998, pp 685-702.
[14] X.P. Liu and C.S. Celikovsky, "Feedback Control of Affine Nonlinear Singular Control Systems", *Int. J. of Control*, vol. 68, 1997, pp 753-774.
[15] X.P. Liu and D. W. C. Ho, "Global Regularization of Nonlinear Differential-Algebraic Equation Systems", *Proc. of the 6th World Congress on Intelligent Control and Automation*, pp 863-867, 2006.
[16] X.P. Liu and D. W. C. Ho, "Stabilization of Nonlinear Differential-Algebraic Equation Systems", *Int. J. of Control*, vol. 77, 2004, pp 671-684.
[17] N.H. McClamroch, "Feedback Stabilization of Control Systems Described by a Class of Nonlinear Differential-Algebraic Equations", *Systems& Control Letters*, vol. 15, 1990, pp 53-60.
[18] R.W. Newcomb, "The semistate description of nonlinear time-viarable circuits", *IEEE Tans. Circuits & Systems*, vol. 28, 1981, pp 61-71.
[19] A. Rehm and F. Allgower, "An LMI approach towards general quadratic performance analysis andsynthesis of descriptor systems" *Proc. of American Control Conference*, Vol. 2, pp 1299-1303, 1999.
[20] D. W. Wang and Y. C. Soh, "Control of Constrained Manipulator with Flexible Joints", *Dynamics and Control*, vol. 6, 1996, pp 33-48.
[21] L.S. You and B.S. Chen, "Tracking Control Designs for Both Holonomic and Non-Holonomic Constrained Menchanical Systems: a Unified Viewpoint", *Int. J. of Control*, vol. 58, 1993, pp 587-612.
[22] G. Zimmer and J. Meier, "On Observing Nonlinear Descriptor Systems", *Systems & Control Letters*, vol. 32, 1997, pp 43-48.

Appendix

Proof of Lemma 2. By carrying out Algorithms 1-3, the system (1)-(3) is changed to the equivalent system (30)-(31). The matrix W takes the form of

$$U\begin{bmatrix} \frac{\partial\varepsilon}{\partial x} & \frac{\partial\xi}{\partial x} \end{bmatrix}^T \quad (44)$$

where U is given in (46) for $i = 0, 1, \cdots, q^k - 1$ and $k = 1, \cdots, m$. Since the matrix $\begin{bmatrix} \frac{\partial\varepsilon}{\partial x} & \frac{\partial\xi}{\partial x} \end{bmatrix}^T$ is nonsingular, therefore the rank of the matrix U is the same as that of the matrix W for $i = 0, 1, \cdots, q^k - 1$ and $k = 1, \cdots, m$. In the matrix U, $\frac{\partial\phi(x)}{\partial\varepsilon} = I$, $\frac{\partial\phi(x)}{\partial\xi} = 0$,

$$\frac{\partial\psi_i^k(x)}{\partial\xi_j^l} = \begin{cases} 1, & k = l \text{ and } i = j \\ 0, & k \neq l \text{ or } i \neq j \end{cases}$$

Since the rank of the matrix W or U is $r + \sum_{j=1}^{k-1} q^j + i + 1$, $\frac{\partial\varphi_i^k(x)}{\partial\xi_j^l} = 0$ for $l > k$ and $i > j$ if $l = k$, which implies that φ_i^k is the function of $\varepsilon, \xi^1, \cdots, \xi^{k-1}, \xi_0^k, \cdots, \xi_i^k$. Therefore, the system (30)-(31) can be expressed by (32)-(33).